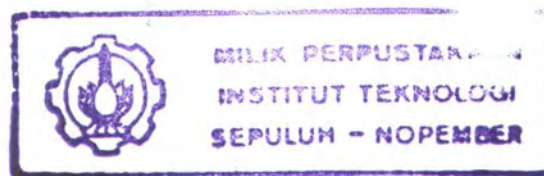
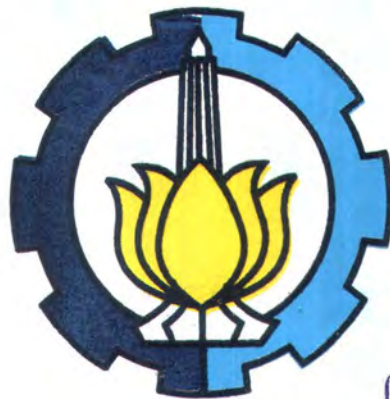


22018/H/05



**PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK
SISTEM INFORMASI PASIEN DAN REKAM MEDIK
DI RUMAH SAKIT ISLAM DARUS SYIFA'
BENOWO SURABAYA**

TUGAS AKHIR



RSIF
015.1
Goe
p-1
2005

OLEH :

HENDRO GOENAWAN

5197 100 044

PERPUSTAKAAN ITS	
Tgl. Terima	5-4-2005
Terima Dari	H
No. Agenda Prp.	221739

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNOLOGI INFORMASI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
2005**

**PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK
SISTEM INFORMASI PASIEN DAN REKAM MEDIK
DI RUMAH SAKIT ISLAM DARUS SYIFA'
BENOWO SURABAYA**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan
Untuk Memperoleh Gelar Sarjana Komputer
Pada
Jurusan Teknik Informatika
Fakultas Teknologi Informasi
Institut Teknologi Sepuluh Nopember
Surabaya**

Mengetahui / Menyetujui

Dosen Pembimbing I


Ir. Aris Tjahyanto, M.Kom
NIP. 131 933 299

Dosen Pembimbing II


Imam Kuswardayan, S.Kom
NIP. 132 306 543

**SURABAYA
JANUARI, 2005**

ABSTRAK

Sistem Informasi Pasien dan Rekam Medik adalah sistem yang dapat mencatat segala aktifitas pasien yang dimulai dari pendaftaran di rumah sakit, perawatan pasien di ruangan perawatan sampai dengan pasien pulang dari rumah sakit.

Sistem Informasi Pasien dan Rekam Medik sangat dibutuhkan oleh rumah sakit karena sangat berhubungan dengan proses pelayanan pasien di rumah sakit terutama dalam kecepatan pelayanan terhadap pasien, proses pengolahan data pasien dan rekam medik dan pembuatan laporan-laporan rumah sakit.

Dalam pembuatan Sistem Informasi Pasien dan Rekam Medik digunakan metodologi System Development Life Cycle (SDLC) dengan menggunakan metode Rapid Application Development (RAD) dapat mengembangkan aplikasi secara terstruktur yang dimulai dari tahap "Project Identification and Selection" sampai tahap "Maintenance".

Dari uji coba terhadap aplikasi Sistem Informasi Pasien dan Rekam Medik yang telah dibuat, didapat hasil bahwa aplikasi dapat menjalankan seluruh fungsi yang ada padanya dengan baik dan telah memenuhi kebutuhan untuk mencatat segala aktifitas pasien ketika di rumah sakit.

KATA PENGANTAR

Dengan mengucapkan rasa syukur kehadirat Allah SWT atas segala rahmat-Nya, sehingga penulis dapat menyelesaikan tugas akhir yang berjudul:

**“PERANCANGAN DAN PEMBUATAN PERANGKAT LUNAK
SISTEM INFORMASI PASIEN DAN REKAM MEDIK
DI RUMAH SAKIT ISLAM DARUS SYIFA’ BENOWO SURABAYA”**

Tugas akhir ini disusun sebagai salah satu persyaratan akademis dengan beban 4 sks untuk memperoleh gelar sarjana pada Jurusan Teknik Informatika, Fakultas Teknologi Informasi, Institut Teknologi Sepuluh Nopember, Surabaya.

Penulis berharap Tugas Akhir ini dapat bermanfaat untuk ilmu pengetahuan dan bahan masukan bagi pembaca dalam bidang Pembuatan Sistem Informasi Pasien dan Rekam Medik di Rumah Sakit, khususnya bagi penulis sendiri. Amin.

Surabaya, Januari 2005

Penulis

DAFTAR ISI

ABSTRAK.....	i
KATA PENGANTAR.....	ii
UCAPAN TERIMA KASIH.....	iii
DAFTAR ISI.....	iv
DAFTAR GAMBAR.....	vi
DAFTAR TABEL.....	viii
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang.....	1
1.2 Permasalahan.....	2
1.3 Batasan Masalah.....	3
1.4 Tujuan.....	4
1.5 Metodologi.....	5
1.6 Sistematika Pembahasan.....	6
BAB II TEORI PENUNJANG.....	7
2.1 Pengenalan Client / Server Database Computing.....	7
2.1.1 Mainframe Database Computing.....	7
2.1.2 PC/File Server Database Computing.....	8
2.1.3 Client/Server Database Computing.....	9
2.1.4 Contoh-Contoh Client / Server Database.....	10
2.2 Borland Delphi.....	11
2.2.1 User Interface.....	12
2.2.2 Modul Data.....	13
2.2.3 Data Source.....	13
2.2.4 Koneksi Data.....	14
2.3 PostgreSQL.....	14
2.3.1 Arsitektur PostgreSQL.....	15
2.4 Dasar-Dasar SQL.....	16
2.5 Konsep Sistem Informasi Pasien dan Rekam Medik.....	17
2.5.1 Rumah Sakit.....	17
2.5.1.1 Tugas dan Fungsi Rumah Sakit.....	18
2.5.1.2 Jenis Pelayanan Medis Rumah Sakit.....	19
2.5.2 Sistem Informasi Pasien dan Rekam Medik.....	21
2.5.2.1 Sistem Informasi Pasien.....	21
2.5.2.2 Pengertian Rekam Medik.....	22
2.5.2.3 Kegunaan Rekam Medik.....	22
2.5.2.4 Kelengkapan Rekam Medik.....	23
2.6 Rumah Sakit Islam Darus Syifa' Benowo Surabaya.....	24
2.6.1 Letak Geografis RSI Darus Syifa' Benowo Surabaya.....	24
2.6.2 Sejarah RSI Darus Syifa' Benowo Surabaya.....	24
2.6.3 Tujuan Pembangunan RSI Darus Syifa' Benowo Surabaya.....	25
2.6.4 Ruang Perawatan di RSI Darus Syifa' Benowo Surabaya.....	26
2.6.5 Existing Sistem di RSI Darus Syifa' Benowo Surabaya.....	28
BAB III KEBUTUHAN SISTEM.....	31
3.1 Metodologi Mencari Kebutuhan Sistem.....	31

3.2	Hasil-Hasil yang Didapatkan.....	33
3.3	Ringkasan Kebutuhan Sistem.....	41
BAB IV ANALISA DAN DESAIN SISTEM		50
4.1	Analisa Sistem.....	50
4.1.1	Proses Bisnis.....	50
4.1.2	Pembuatan Conceptual Data Model	67
4.2	Desain Sistem	67
4.2.1	Files dan Database.....	69
4.2.2	Desain Forms.....	74
4.2.3	Desain Reports	87
4.2.4	Desain Menu Sistem.....	91
4.2.5	Spesifikasi Proses	91
BAB V IMPLEMENTASI DAN UJI COBA SISTEM		99
5.1	Implementasi Sistem	99
5.1.1	SUB_DBMODULE.....	100
5.1.2	SUB_DBMODULE_INVOICE	102
5.1.3	SUB_DBMODULE_JOURNAL	105
5.1.4	SUB_DBMODULE_LEASING	105
5.1.5	SUB_DBMODULE_OBJECT.....	106
5.1.6	SUB_DBMODULE_OWNER.....	108
5.1.7	SUB_DBMODULE_PROGRAM.....	114
5.1.8	SUB_DBMODULE_TRANSACT	116
5.1.9	SUB_DBMODULE_MEDICAL	119
5.2	Uji Coba Sistem.....	121
5.2.1	Lingkungan Uji Coba	121
5.2.2	Skenario Uji Coba	122
5.2.3	Pelaksanaan Uji Coba.....	122
5.2.3.1	Pengaturan Konfigurasi server PostgreSQL.....	123
5.2.3.2	Login User.....	124
5.2.3.3	Uji Coba Pelayanan Rawat Jalan.....	125
5.2.3.4	Uji Coba Pelayanan Rawat Inap.....	131
5.2.3.5	Uji Coba Pelayanan Apotek Umum	139
5.2.3.6	Uji Coba Pelayanan Pembayaran Hutang	140
5.2.3.7	Uji Coba Pencetakan Laporan Rumah Sakit	141
5.2.4	Hasil Uji Coba	141
BAB VI PENUTUP		142
6.1	Kesimpulan.....	142
6.2	Kemungkinan Pengembangan	143
DAFTAR PUSTAKA		144
LAMPIRAN.....		145

DAFTAR GAMBAR

Gambar 2.1 Mainframe Database Computing	8
Gambar 2.2 PC/File Server Database Computing	9
Gambar 2.3 Client/Server Database Computing	10
Gambar 2.4 Desain umum aplikasi database	12
Gambar 2.5 Data Modul	13
Gambar 2.6 Data Source	13
Gambar 2.7 Struktur Organisasi RSI Darus Syifa' Benowo Surabaya	26
Gambar 2.8 Alur Pasien Rumah Sakit	30
Gambar 3.1 Rapid Application Development	31
Gambar 4.1 System Context Diagram Sistem Informasi Pasien dan Rekam Medik RSI Darus Syifa' Benowo Surabaya	58
Gambar 4.2 Level 0 DFD Sistem Informasi Pasien dan Rekam Medik RSI Darus Syifa' Benowo Surabaya	59
Gambar 4.3 Level 1 Proses Rawat Jalan	60
Gambar 4.4 Level 2 Proses Tindakan dan Rekam Medik Rawat Jalan	61
Gambar 4.5 Level 1 Proses Rawat Inap	62
Gambar 4.6 Level 2 Proses Tindakan dan Rekam Medik Rawat Inap	63
Gambar 4.7 Level 1 Proses Apotik	64
Gambar 4.8 Level 1 Proses Pembayaran	65
Gambar 4.9 Level 1 Proses Hutang Pasien	66
Gambar 4.10 Conceptual Data Model	68
Gambar 4.11 Form User Login	78
Gambar 4.12 Form User Manager	78
Gambar 4.13 Form Input Pasien Baru	79
Gambar 4.14 Form Input Pasien Rawat Jalan	79
Gambar 4.15 Form Input Pasien Rawat Inap	80
Gambar 4.16 Form Input Pasien Pindah Kamar	80
Gambar 4.17 Form Input Rekam Medik Pasien	81
Gambar 4.18 Form Input Transaksi Tindakan Pasien	82
Gambar 4.19 Form Input Pembayaran Transaksi Tindakan Pasien	82
Gambar 4.20 Form Input ICD Pasien	83
Gambar 4.21 Form Input Transaksi Obat Pasien	84
Gambar 4.22 Form Pembayaran Pulang Rawat Inap	85
Gambar 4.23 Form Statistik Rumah Sakit	86
Gambar 4.24 Desain Laporan DaftarPasienRJ	88
Gambar 4.25 Desain Laporan MorbiditasReportRI	89
Gambar 4.26 Desain Laporan RekapKarcisRawatJalan	89
Gambar 4.27 Desain Laporan TransactionByCustomer	90
Gambar 4.28 Desain Laporan PoliReportDiagnosisDokter	90
Gambar 4.29 Desain Menu Root Aplikasi	91
Gambar 5.1 Form Login User	124
Gambar 5.2 Form Main Menu	124
Gambar 5.3 Form Daftar Pasien Rawat Jalan	125

Gambar 5.4 Form Input Pasien Rawat Jalan.....	126
Gambar 5.5 Form Pasien Rawat Jalan Updated.....	126
Gambar 5.6 Kwitansi Karcis Rawat Jalan	127
Gambar 5.7 Daftar Pasien Poliklinik Gigi.....	127
Gambar 5.8 Form MedRec Free Form.....	128
Gambar 5.9 Form Rekam Medik Pasien	129
Gambar 5.10 Laporan Rekam Medik Pasien.....	129
Gambar 5.11 Form Transaksi Tindakan Pasien.....	130
Gambar 5.12 Form Pembayaran Tindakan Pasien	130
Gambar 5.13 Kwitansi Pembayaran Transaksi	131
Gambar 5.14 Form Pendaftaran Pasien Rawat Inap.....	131
Gambar 5.15 Form Input Pasien Rawat Inap	132
Gambar 5.16 Form Input Pasien Rawat Inap	132
Gambar 5.17 Form Pasien Rawat Inap Updated	133
Gambar 5.18 Kwitansi Uang Muka Rawat Inap.....	133
Gambar 5.19 Daftar Pasien Ruangan Perawatan.....	134
Gambar 5.20 Form Summary Rekam Medik	135
Gambar 5.21 Form Summary Transaksi Tindakan.....	135
Gambar 5.22 Form Transaksi Tindakan Pasien Rawat Inap	136
Gambar 5.23 Form Pemakaian Kamar	136
Gambar 5.24 Form Transaksi Obat Pasien Rawat Inap	137
Gambar 5.25 Form Transaksi Obat Pasien Rawat Inap	138
Gambar 5.26 Form Pembayaran Pulang Rawat Inap	138
Gambar 5.27 Form Penjualan Obat Umum.....	140
Gambar 5.28 Form Pembayaran Hutang	141

DAFTAR TABEL

Tabel 2.1 Jumlah Kamar RSI Darus Syifa' Benowo Surabaya	27
Tabel 4.1 Daftar Table dalam Database	69
Tabel 4.2 Daftar Form.....	74
Tabel 4.3 Daftar Laporan	87
Tabel 5.1 Daftar DBModule	99

BAB I

PENDAHULUAN

1.1 Latar Belakang

Dinamika masyarakat yang berkembang di bidang sosial budaya, ilmu pengetahuan, teknologi dan industri menjadikan suatu keadaan yang senantiasa menuntut suatu pelayanan prima, yaitu pelayanan yang tepat, cepat, terjangkau dan memuaskan pelanggan. Perkembangan dan tuntutan seperti ini juga terjadi di bidang kesehatan. Rumah Sakit merupakan salah satu tiang dalam perkembangan pelayanan kesehatan, sudah barang tentu juga membutuhkan sarana dan prasarana dalam mewujudkan pelayanan prima.

Rumah sakit merupakan sistem pelayanan kesehatan yang sifatnya massal, hal ini dapat kita lihat dari banyaknya pasien yang dirawat, banyaknya dokter dan staff yang bertugas didalamnya yang dikelompokkan dalam departemen berdasarkan bidang pelayanan. Dari banyaknya orang-orang yang terlibat didalamnya dibutuhkan suatu sistem yang dapat mengorganisir serta mengatur secara cepat, akurat, mudah dan efisien guna mendukung peningkatan pelayanan di rumah sakit.

Sistem Informasi merupakan salah satu umpan balik untuk management dalam merencanakan, memutuskan kebijakan apa yang akan dikerjakan. Sistem Informasi disini dituntut untuk memberikan solusi dari berbagai aspek, seperti

aspek medis, aspek keuangan, dan aspek lain yang terkait secara langsung maupun tidak langsung terhadap kelangsungan pelayanan rumah sakit itu sendiri.

Pada umumnya sistem informasi di rumah sakit dapat dikelompokkan lagi ke dalam sistem yang lebih kecil antara lain

1. Sistem Informasi Pasien,
2. Sistem Rekam Medik.
3. Sistem Keuangan,
4. Sistem Staff Management,
5. Sistem Informasi Barang dan Jasa,
6. Sistem Asset Management.

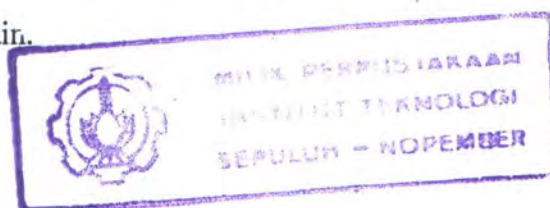
Sistem Informasi Pasien di dalam rumah sakit adalah suatu sistem yang kritis di dalam suatu rumah sakit, karena sangat berhubungan dengan kecepatan pelayanan terhadap pasien di rumah sakit, sebab keterlambatan pelayanan terhadap pasien akan berakibat fatal bagi pasien tersebut dan juga terhadap nama baik suatu rumah sakit.

Sistem Rekam Medik digunakan untuk melakukan pencatatan segala jenis tindakan dokter terhadap pasien yang sesuai dengan aturan-aturan yang berlaku. Dengan adanya Rekam Medik memudahkan dokter membuat keputusan dengan melihat kembali hasil pemeriksaan pasien terdahulu.

1.2 Permasalahan

Parameter-parameter yang diangkat menjadi permasalahan dalam pengerjaan tugas akhir ini adalah :

- Bagaimana merancang dan membuat suatu Sistem Informasi yang dapat mengatur aliran pasien serta segala macam transaksi dan pemeriksaannya yang ada di Rumah Sakit Islam Darus Syifa' yang juga dapat mengintegrasikan setiap departemen di Rumah Sakit Islam Darus Syifa', dimana disetiap departemen mempunyai transaksi-transaksi yang identik tapi tidak sama.
- Bagaimana sistem informasi yang dirancang dapat mencatat segala tindakan medis dari dokter terhadap pasien dalam rekam medik yang dapat dilihat kembali oleh dokter ketika pasien berobat kembali.
- Bagaimana sistem informasi yang dirancang dan dibuat dapat melakukan pengelolaan user sesuai dengan otoritas yang diberikan oleh pihak rumah sakit di dalam merancang dinamik menu yang diintegrasikan dengan user login, dimana tiap user login akan mempunyai modul-modul yang berbeda dengan user login yang lain sesuai dengan otoritas atau hak akses yang diberikan oleh pihak rumah sakit.
- Bagaimana sistem informasi yang dirancang dapat membuat laporan-laporan untuk Rumah Sakit, antara lain Jurnal Harian, Statistik Rumah Sakit, Laporan Morbiditas Rumah Sakit dan lain-lain.



1.3 Batasan Masalah

Adapun batasan masalah dalam tugas akhir ini adalah:

- Perangkat lunak yang akan dirancang hanya berhubungan dengan segala macam kegiatan pasien di rumah sakit mulai dari pasien masuk rumah sakit sampai pasien pulang dari rumah sakit. Tidak mencakup pengaturan stok oleh

apotik, administrasi keuangan rumah sakit, dan yang tidak berhubungan dengan pasien.

- Bahasa pemrograman yang akan digunakan dalam pembuatan perangkat lunak sistem informasi ini adalah Borland Delphi untuk mendapatkan performa yang cepat dalam melakukan transaksi-transaksi dan pembuatan laporan-laporan. Sedangkan DBMS yang digunakan adalah PostgreSQL karena dibutuhkan DBMS dengan kinerja yang sekelas ORACLE dengan dana yang minimal.

1.4 Tujuan

Adapun tujuan dalam tugas akhir ini adalah:

- Didapatkannya rancangan Sistem Informasi Pasien serta Rekam Medik yang dapat mendukung peningkatan produktivitas seluruh kegiatan yang ada di Rumah Sakit Islam Darus Syifa' sehingga didapatkan hasil yang optimal baik dari segi efisiensi biaya dan segi waktu.
- Optimalisasi segala sumber daya di Rumah Sakit Islam Darus Syifa' khususnya yang berhubungan dengan pasien. Kemudian Sistem Informasi tersebut diharapkan dapat mengintegrasikan semua departemen-departemen yang ada di Rumah Sakit Islam Darus Syifa' secara menyeluruh sehingga diharapkan tidak akan terjadi informasi dan komunikasi yang terputus antar bagian yang terlibat.
- Tersedianya kemampuan konsolidasi data dan informasi dari seluruh bidang di Rumah Sakit Islam Darus Syifa' sehingga pihak manajemen rumah sakit dapat dengan mudah melakukan analisa dan membuat keputusan.

1.5 Metodologi

Metodologi yang digunakan dalam pembuatan tugas akhir ini adalah sebagai berikut :

a. Studi literatur

Mencari, mempelajari dan merangkum berbagai macam literatur yang berkaitan dengan rumusan masalah, teori-teori yang berhubungan dengan sistem yang akan dibangun.

b. Pengumpulan dan analisis data

Melakukan survey di Rumah Sakit Islam Darus Syifa' Benowo untuk mengetahui kebutuhan-kebutuhan user dan mendapatkan data-data yang akan digunakan dalam sistem yang akan dibangun.

c. Perancangan desain dan implementasi Sistem

Merancang proses-proses bisnis yang terdapat di rumah sakit ke dalam bentuk System Context Diagram, DFD-DFD hasil dari Decompose proses-proses dan Entity Relationship Diagram dalam bentuk CDM atau PDM dari data-data yang didapatkan dari hasil survey. Kemudian diimplementasikan ke dalam bahasa pemrograman Borland Delphi sehingga menjadi aplikasi yang diinginkan.

d. Uji coba Sistem

Melakukan uji coba perangkat lunak Sistem Informasi Pasien dan Rekam Medik di Rumah Sakit Islam Darus Syifa', untuk mengetahui bahwa aplikasi sudah berjalan sesuai dengan tujuan yang diharapkan dalam pembuatan aplikasi Sistem Informasi ini.

e. Penyusunan laporan

Penyusunan buku sebagai dokumentasi dari pelaksanaan Tugas Akhir.

1.6 Sistematika Pembahasan

Pembahasan dalam Tugas Akhir ini dibagi menjadi beberapa bab sebagai berikut :

- Bab I, Pendahuluan, berisi latar belakang, permasalahan, tujuan, batasan masalah, metodologi dan sistematika pembahasan.
- Bab II, Teori Penunjang, akan dibahas dasar ilmu yang mendukung pembuatan Tugas Akhir ini seperti konsep pemrograman *client/server*, Borland Delphi, dan PostgreSQL. Serta dibahas konsep dasar Sistem Informasi Pasien dan Rekam Medik dan gambaran dari Rumah Sakit Islam Darus Syifa' Benowo Surabaya.
- Bab III, Metodologi Pembuatan Sistem. Pada bab ini akan dibahas tentang langkah-langkah dalam pembuatan sistem informasi dan hasil-hasil yang telah didapatkan.
- Bab IV, Analisa dan Desain Sistem. Pada bab ini akan dibahas mengenai Analisa dan Desain Sistem yang sesuai dengan hasil-hasil yang didapatkan dari Bab III.
- Bab V, Implementasi dan Uji Coba Sistem, Pada bab ini akan dilakukan Implementasi dan Uji Coba Sistem dari hasil Analisa dan Desain diatas.
- Bab VI, Penutup, berisi kesimpulan yang dapat diambil dari Tugas Akhir ini beserta saran untuk pengembangan selanjutnya.

BAB II

TEORI PENUNJANG

Dalam bab ini dibahas dasar teori yang menunjang pembuatan Tugas Akhir. Pertama kali akan dibahas tentang konsep Client / Server Database. Selanjutnya dibahas tentang konsep pemrograman dengan menggunakan Borland Delphi dan mengenai PostgreSQL DBMS. Dan pembahasan terakhir mengenai konsep Sistem Informasi Pasien dan Rekam Medik serta deskripsi dari Rumah Sakit Islam Darus Syifa' Benowo.

2.1 Pengenalan Client / Server Database Computing

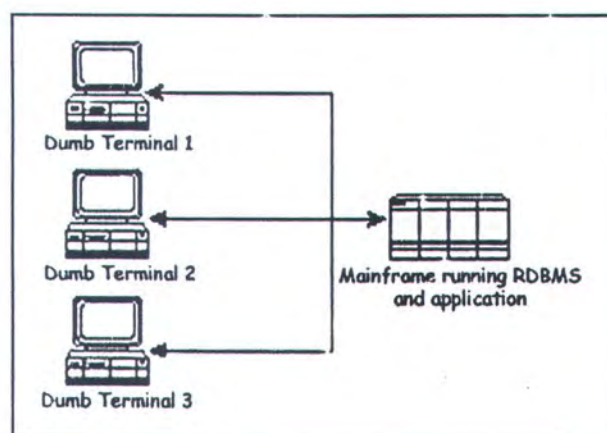
Client/server (C/S) database computing tergolong teknologi baru, terlebih untuk lingkungan jaringan dengan PC. Perkembangannya tidak terlepas dari kebutuhan akan model penanganan database yang baru selain alternatif yang sudah ada, seperti database pada mainframe dan database PC/File Server [8].

2.1.1 Mainframe Database Computing

Sebelum tahun 80-an dan awal 90-an, *mainframe computing* merupakan satu satunya pilihan untuk organisasi dengan pemrosesan yang tinggi dan banyak user. Selama kurun waktu 20 tahunan, mainframe dapat membuktikan reliabilitas dan kemampuan untuk mendukung pemrosesan (termasuk terhadap database) oleh banyak user secara serentak.

Model seperti ini terdiri dari mainframe dan sejumlah dumb terminal (yang berfungsi untuk input/ output). Model ini juga dikenal dengan *host-based computing*, merefer pada semua pemrosesan dilakukan pada mainframe. Dumb terminal hanya berfungsi sebagai tempat untuk menerima masukan dan menampilkan hasil (format layar untuk input/ output ditangani oleh mainframe). Sebagai contoh, mainframe menjalankan RDBMS, mengatur aplikasi yang mengakses RDBMS tersebut dan menangani komunikasi antara mainframe dan dumb terminal.

Kekurangan dari mainframe computing adalah mahal, karena semua menggunakan peralatan yang tidak umum.



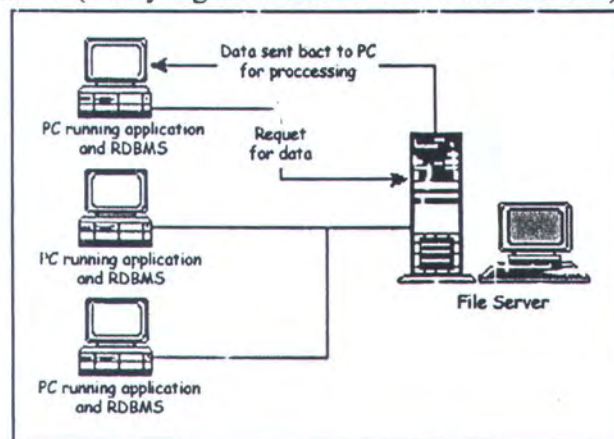
Gambar 2.1 Mainframe Database Computing

2.1.2 PC/File Server Database Computing

PC/file server-based computing terkenal pada akhir tahun 80-an saat dikenalnya PC. User menyukai komputer jenis ini karena kemudahan pengoperasian dan kemampuan yang cukup untuk memenuhi kebutuhan. Hal ini meningkat dengan diperkenalkannya sistem operasi LAN yang mendukung file server ditambah kemampuan multi user pada aplikasi yang dibuat.

Pemrosesan tidak lagi dilakukan dipusat (mainframe pada host based computing), tetapi pada tiap station yang menjalankan aplikasi. Komputer yang bertindak sebagai file server berfungsi untuk penyimpanan file.

Kekurangan dari model ini justru terletak pada ketidakpedulian file server terhadap proses yang dilakukan. File server membagikan data yang diperlukan untuk diproses di station (PC lokal), tanpa melakukan proses apa-apa. Ini menyebabkan ketidakefisienan dan kesibukan lalu lintas data pada media komunikasi (data yang dilewatkan adalah data mentah).



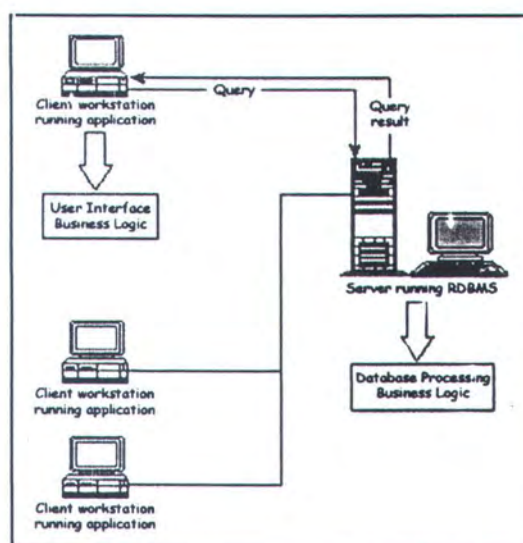
Gambar 2.2 PC/File Server Database Computing

2.1.3 Client/Server Database Computing

Sistem client/server pada mulanya dirancang untuk menghasilkan kinerja yang jauh lebih besar dengan hanya menambah sedikit biaya, dengan cara memindahkan sebagian tugas pemrosesan dari komputer kepada server.

Gambar 2.3 menunjukkan bagaimana konfigurasi sistem client/server secara umum. Pada gambar, komputer menjalankan permintaan SQL, permintaan itu dikirim melalui jaringan kepada sever Database. Server Database kemudian memproses permintaan itu dan mengembalikan data

kepada komputer client. Operasi sistem client/server yang ditunjukkan di sini berbeda dari aplikasi database pada Shared Database dalam cara bagaimana data diproses. Jika memakai program Shared Database seperti Access, seluruh tabel database yang diminta akan dikirim kepada komputer client untuk diproses. Komputer client kemudian akan memproses data itu dan menampilkan hasilnya. Kinerja dari Client akan menurun jika melakukan hal ini karena jumlah data yang dikirim melalui network dan beban aplikasi tidak dibagi antara Client dan Server. Satu-satunya tugas yang dilakukan oleh server dalam hal ini ialah mengatur pembagian file kepada PC yang memintanya.



Gambar 2.3 Client/Server Database Computing

2.1.4 Contoh-Contoh Client / Server Database

Client /Server Database itu banyak sekali macamnya. Setiap client / server database tersebut mempunyai kelebihan serta kekurangannya masing-masing. Berikut ini akan diberikan penjelasan singkat mengenai client / server database yang banyak digunakan atau dipakai oleh user.

- **MS SQL**

MS SQL adalah sebuah Client/Server Database yang banyak digunakan di perusahaan-perusahaan. Kelebihan dari MS SQL ini adalah user interface yang bagus sehingga memudahkan user menggunakannya, serta kapasitas data yang mampu disimpan cukup besar.

- **ORACLE**

ORACLE adalah Client/Server database yang terkenal ketangguhannya, hal ini dibuktikan dengan banyak dipakainya ORACLE pada perusahaan besar. Selain itu kapasitas dari ORACLE juga sangat besar sampai bergiga-giga, selain itu ORACLE dapat dibuat sebagai parallel database.

- **MYSQL**

Mysql adalah sebuah client / Server database yang sangat free (gratis) serta open source, selain itu mysql dalam mengeksekusi suatu query SQL lebih cepat dibanding yang lain. Kelemahan mysql dibanding yang lain adalah mysql DBMS bukan RDBMS.

- **POSTGRESQL**

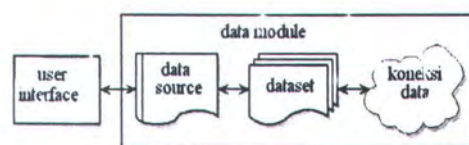
Kelebihan PostgreSQL dibanding yang lain adalah free (gratis) serta open source dan mempunyai fasilitas lengkap (mendukung inheritance table, object model, dll).

2.2 Borland Delphi

Dengan semakin berkembangnya sistem operasi Windows, Borland International, sebuah perusahaan software Amerika, mengeluarkan versi Windows

untuk bahasa pemrograman Pascal yaitu Borland Delphi. Sebagaimana umumnya bahasa pemrograman di dalam sistem operasi Windows, Delphi juga bahasa pemrograman yang berorientasi pada obyek (Object Oriented Programming), sehingga mendukung sekali sebagai Rapid Application Developer (RAD- Pengembang Aplikasi yang cepat). Hal ini dikarenakan Delphi membagi struktur batang programnya di dalam unit-unit, dimana kita bisa menggunakan kembali unit-unit ini di dalam project yang berbeda [1].

Secara umum aplikasi database terdiri dari dua bagian : user interface dan koneksi ke database itu sendiri. User interface merupakan komponen yang menyajikan informasi data, sedangkan koneksi ke database adalah komponen yang menyambung antara user interface dengan database. Cara mengorganisasikan kedua elemen ini tergantung dari arsitektur aplikasi masing-masing. Secara umum gambaran aplikasi database adalah sebagai berikut:



Gambar 2.4 Desain umum aplikasi database

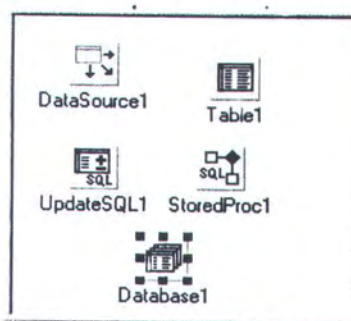
2.2.1 User Interface

Desain yang baik dari sebuah aplikasi database adalah memisahkan *user interface* dengan database sehingga betul-betul terpisah. Keuntungan dari pemisahan ini adalah fleksibilitas program yaitu *user interface* tidak perlu dimodifikasi jika suatu saat kita harus memodifikasi database dan begitu juga sebaliknya. Juga pemisahan semacam ini dapat membuat kita

mengembangkan aplikasi dengan cepat, karena ketika kita membuat aplikasi baru, *user interface* cukup mengambil *user interface* aplikasi yang lama.

2.2.2 Modul Data

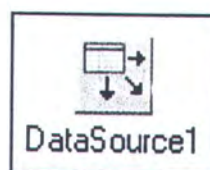
Modul Data ini digunakan sebagai kontainer yang berisi komponen-komponen database (dataset), dan komponen koneksi ke dataset tersebut dari bagian aplikasi lainnya misalnya *user interface*.



Gambar 2.5 Data Modul

2.2.3 Data Source

Komponen pertama dalam modul data adalah data source. *Datasource* ini bertindak sebagai penyambung antara *user interface* dan sebuah dataset yang berisi informasi dari database. Beberapa komponen database dalam sebuah form dapat berbagi sebuah *datasource* bersama-sama.



Gambar 2.6 Data Source

Jantung dari aplikasi database delphi adalah dataset. Komponen ini berisi sebuah set record dari database. Record-record ini dapat berasal dari

sebuah tabel, sebuah subset pada record, atau lebih dari satu tabel. Dengan menggunakan dataset, aplikasi tidak harus merubah struktur programnya jika tabel databasenya secara fisik berubah

2.2.4 Koneksi Data

Dasar dari koneksi database Delphi adalah dengan ODBC (Open Database Connectivity) melalui perantara ADO Connection. Dimana ODBC yang secara langsung menghubungkan antara ADO dengan DBMS.

2.3 PostgreSQL

PostgreSQL merupakan Database Management System (DBMS) server open source. PostgreSQL memberikan dukungan untuk Object Relational (ORDBMS), dan didukung oleh sebuah team pengembang yang melakukan pekerjaannya dengan komunikasi lewat Internet. PostgreSQL merupakan turunan *open source* dari kode asli Berkeley yang menyediakan dukungan bahasa SQL92/SQL99 serta fitur modern lainnya [4].

Ada beberapa jalan untuk melakukan pengukuran suatu perangkat lunak, yaitu fitur, kinerja, reliabilitas, dukungan, dan harga. Penjelasannya sebagai berikut :

- Fitur: PostgreSQL menawarkan fitur-fitur yang ada pada produk DBMS komersial, seperti *transaction*, *trigger*, *view*, *foreign key referential integrity*, dan *sophisticated locking*. Juga terdapat fasilitas *user-defined type*, *inheritance*, *rule*, dan *multi-version concurrency control* untuk mengurangi *lock contention*.

- Kinerja: Setiap koneksi user ditangani dengan membuat proses Unix. Proses *backend* akan melakukan share buffer data dan mengunci informasi. Dengan multiple CPU, multiple backend bisa berjalan dengan mudah pada CPU yang berbeda.
- Reliabilitas: Suatu DBMS harus mempunyai reliabilitas. Kode yang di-release harus stabil dan memiliki jumlah bug yang minimal. Setiap release sekurang-kurangnya mengalami sebulan beta testing, dan selama itu telah nampak release yang dipergunakan.
- Dukungan: Mailing list yang memungkinkan sejumlah besar developer dan user menangani masalah yang timbul. DBMS komersial sendiri juga tidak selalu memungkinkan suatu perbaikan. Akses langsung ke developer, komunitas user, manual dan source code membuat dukungan PostgreSQL cukup superior dibanding DBMS lain.
- Harga: PostgreSQL gratis untuk semua pemakaian, baik komersial maupun tidak. Kita bisa pula menambahkan kode tanpa adanya batasan, kecuali pada bagian yang dinyatakan dalam style BSD.

2.3.1 Arsitektur PostgreSQL

Dalam istilah database, PostgreSQL mempergunakan model client-server sederhana "process per-user" [4]. Suatu session Postgres terdiri dari proses (program) Unix yang saling bekerjasama berikut:

- Suatu proses supervisory daemom. (postmaster)
- Aplikasi frontend user (misal program pgsqll)

- Satu atau lebih backend database server (proses postgres sendiri).

Postmaster mengelola sejumlah database pada satu host. Kumpulan database tersebut disebut sebagai cluster (dari database). Aplikasi frontend yang akan mengakses database dalam suatu cluster membuat panggilan ke library. Library mengirimkan permintaan user melalui jaringan ke postmaster, yang akan mengaktifkan proses server backend baru, dan menghubungkan proses frontend ke server baru. Proses frontend juga bisa terkoneksi ke banyak backend server.

2.4 Dasar-Dasar SQL

Bahasa SQL adalah sarana dasar yang dibutuhkan untuk mengakses data di dalam database relasional. Menguasai bahasa SQL adalah langkah pertama dan terpenting di dalam memahami database relasional. Pada beberapa tahun terakhir, bahasa SQL telah meninggalkan domain mainframe dan turun ke tingkat desktop. Perpindahan ini disebabkan karena database relasional telah mengembangkan platform mereka dan saat ini telah memiliki versi-versi yang berjalan baik pada PC maupun mainframe. Selain itu, internet telah merubah menjadi alat yang mempopulerkan bahasa ini. Karena bahasa memiliki struktur yang cocok untuk arsitektur client – server semakin banyak aplikasi atau halaman yang mengakses database relasional yang dibuat dengan menggunakan bahasa SQL [9].

Bahasa SQL merupakan bahasa standar database yang telah menjadi standar bersama oleh semua pengembang perangkat lunak database. Bahasa ini, seperti pemrograman lainnya, merupakan kumpulan perintah-perintah khusus

yang berhubungan dengan record-record yang terdapat pada database, seperti perintah untuk menampilkan, menyisipkan, menghapus atau mengubah record.

SQL berdasarkan fungsinya dibedakan menjadi dua, yaitu DDL (Data Definition Language) serta DML (Data Manipulation Language).

DDL atau Data Definition Language adalah bagian dari SQL yang digunakan untuk mendefinisikan data dan objek database. Contoh DDL dalam PostgreSQL seperti *CREATE TABLE*, *CREATE INDEX*, *ALTER TABLE*, dan *DROP TABLE* [9].

DML atau Data Manipulation Language adalah bagian dari SQL yang digunakan untuk memulihkan atau memanipulasi data. Perintah-perintah ini bertanggung jawab untuk melakukan query dan perubahan yang dilakukan di dalam table. Contoh DML dalam PostgreSQL seperti *SELECT*, *INSERT*, *UPDATE* dan *DELETE* [9].

2.5 Konsep Sistem Informasi Pasien dan Rekam Medik

Dibawah ini akan dibahas mengenai konsep yang mendasari pembuatan Sistem Informasi Pasien dan Rekam Medik.

2.5.1 Rumah Sakit

Rumah Sakit adalah suatu pelayanan kesehatan institusional yang menyelenggarakan kegiatan pelayanan kesehatan :

- kuratif, yaitu pengobatan penyakit
- rehabilitatif, yaitu pemulihan kesehatan



- preventif, yaitu pencegahan penyakit dan
- promotif, yaitu pembinaan kesehatan.

melalui :

- pelayanan medis,
- pelayanan rawat,
- pelayanan mondok,
- pelayanan administratif,

secara :

- rawat jalan, yaitu pada waktu dan jam tertentu,
- rawat darurat, yaitu dalam waktu setinggi-tingginya 24 jam,
- rawat tinggal, yaitu dalam waktu sekurang-kurangnya 24 jam,

dan memiliki sekurang-kurangnya 25 tempat tidur tersedia dan disamping itu dapat menyelenggarakan pendidikan tenaga paramedis, membantu pendidikan tenaga medis, membantu penelitian dan pengembangan kesehatan, dan membantu kegiatan penyelidikan epidemiologi [7].

2.5.1.1 Tugas dan Fungsi Rumah Sakit

Rumah Sakit mempunyai tugas menyelenggarakan pelayanan kesehatan melalui upaya penyembuhan penderita kesehatan melalui upaya pemulihan keadaan gangguan kesehatan badan, jiwa dan ditunjang oleh upaya peningkatan dan upaya pencegahan gangguan kesehatan [7].

Fungsi Rumah Sakit untuk menyelenggarakan tugas tersebut Rumah Sakit yang disesuaikan dengan batasan WHO mempunyai 3

2.5.2.2 Pengertian Rekam Medik

Rekam Medik (Medical Record) diartikan sebagai berikut :

- a. Rekam Medik adalah kumpulan data medis seseorang dimulai dari saat janin hingga berakhir pada saat meninggal,
- b. Rekam Medik dibuat secara kronologis, mencakup keterangan tentang : siapa, apa, mengapa, dimana, kapan, dan bagaimana seorang penderita memperoleh pelayanan medis disuatu unit pelayanan kesehatan,
- c. Rekam Medik merupakan dokumen pengalaman sakit penderita selama berada dalam perawatan dokter, sebaliknya Rekam Medik adalah dokumen resmi suatu unit pelayanan kesehatan untuk pertanggung jawaban pelaksanaan perawatan, terutama di rumah sakit [7].

Rekam Medik juga bisa diartikan sebagai "*Keterangan tertulis dan terekam tentang identitas anamnesa, pemeriksaan fisik laboratorium, diagnosa dan pengobatan*" [6].

2.5.2.3 Kegunaan Rekam Medik

Rekam Medik yang lengkap memiliki fungsi penunjang pelayanan kesehatan. Dirumah sakit ada 3 jenis pelayanan medis yaitu rawat tinggal, rawat jalan, dan rawat darurat [7]. Ketiga hal ini disesuaikan dengan penggolongan kasus penderita yang harus ditangani.

Peranan Rekam Medik adalah besar sekali karena sebagai alat pemersatu serta alat komunikasi antara staff medis dengan staf medis lainnya baik di rumah sakit sendiri maupun di rumah sakit lain dalam rangka usaha penyembuhan penderita.

Secara garis besar kegunaan Rekam Medik adalah sebagaimana berikut :

1. Sebagai alat komunikasi antara dokter dengan tenaga ahli (profesional) lainnya, yang turut ambil bagian dalam usaha memberikan pelayanan, pengobatan, perawatan seorang penderita.
2. Berfungsi sebagai dasar untuk merencanakan pengobatan/perawatan seorang penderita.
3. Menjadi bukti tertulis (documentary evidence) tentang perkembangan penyakit, pengobatan, selama atau tiap kali seorang penderita berkunjung/diawat di rumah sakit.
4. Menjadi bahan yang berguna untuk penelitian, evaluasi dan analisa kualitas pelayanan yang telah diberikan kepada penderita.
5. Melindungi kepentingan hukum (legal interest) bagi penderita, rumah sakit dan dokter.
6. Menyediakan data klinis yang sangat berharga untuk pendidikan medis dan para medis.

2.5.2.4 Kelengkapan Rekam Medik

Rekam Medik menerangkan tentang : Siapa (who), apa (what), mengapa (why), dimana (where), kapan (when) dan bagaimana (how),

seorang penderita memperoleh pelayanan pengobatan dan perawatan selama berhubungan dengan rumah sakit [7].

Dalam Rekam Medik tersimpan data/keterangan tentang penderita dan pelayanan medis yang diperolehnya. Rekam Medik yang lengkap harus berisi keterangan lengkap mengenai : identitas penderita, pendapat yang pasti tentang diagnosa, pengobatan yang diberikan dan hasil-hasilnya.

2.6 Rumah Sakit Islam Darus Syifa' Benowo Surabaya

2.6.1 Letak Geografis Rumah Sakit Islam Darus Syifa' Benowo Surabaya

Secara Geografis Rumah Sakit Islam Darus Syifa' berada di Jalan Raya Benowo No. 05 Benowo Surabaya. Letaknya cukup strategis, yakni terletak diantara perbatasan Surabaya dan Gresik di sebelah kiri jalan dari arah Surabaya, dan dilewati oleh banyak angkutan kota baik dari arah Surabaya maupun Gresik.

2.6.2 Sejarah Rumah Sakit Islam Darus Syifa' Benowo Surabaya

Berawal dari banyaknya warga Benowo yang meninggal dikarenakan kurangnya fasilitas medik di daerah Benowo, maka muncullah keinginan dari muslimat NU kelurahan Benowo untuk membangun "Balai Kesehatan Darus Syifa' Muslimat NU" pada bulan Oktober 1989 yang berada di depan jalan raya benowo (± 100 m di Timur Puskesmas Pembantu) agar seluruh warga kecamatan benowo pada khususnya dan seluruh rakyat pada umumnya mudah

dalam menjangkaunya. Dan dari pada itu kecamatan Benowo terletak dibagian barat kota Surabaya, dari pusat kota berjarak ± 20 km yang terdiri dari 10 kelurahan dan berpenduduk ± 35.000 jiwa.

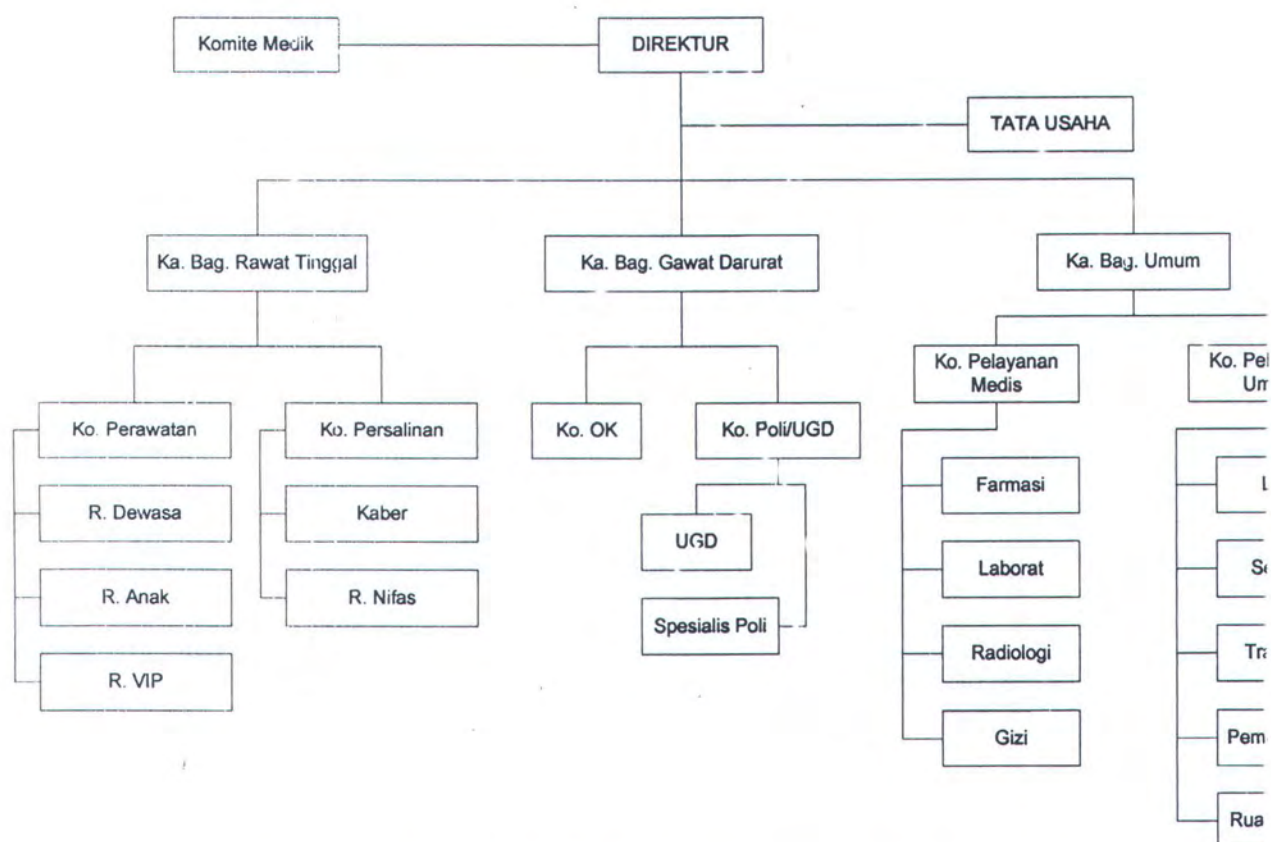
Dengan berkembangnya balai kesehatan ini maka digantilah dengan “Rumah Sakit Islam Darus Syifa’ “ kecamatan Benowo Surabaya yang telah disahkan oleh Akta Notaris A. Kohar, SH nomor 80 tanggal 1 Februari 1990. Akan tetapi operasionalisasi Rumah Sakit Islam ini baru dibuka pertama kali pada tanggal 17 November 1991.

2.6.3 Tujuan Pembangunan Rumah Sakit Islam Darus Syifa’ Benowo Surabaya

Tujuan didirikannya Rumah Sakit Islam Darus Syifa’ Benowo Surabaya yaitu :

1. Agar dapat memenuhi harapan masyarakat dalam kebutuhan kesehatan secara maksimal.
2. Rumah sakit islam ini dibangun oleh jam’iyah NU dan muslimat serta pendiri yang lain untuk memberikan manfa’at pada seluruh masyarakat secara umum, tanpa membedakan kelompok agama, keturunan, politik dan ekonomi.

Struktur organisasi Rumah Sakit Islam Darus Syifa’ Benowo adalah tergambar dalam Gambar 2.7



Gambar 2.7 Struktur Organisasi RSI Darus Syifa' Benowo Surabaya

2.6.4 Ruangan Perawatan di Rumah Sakit Islam Darus Syifa' Benowo Surabaya

Terdapat 10 Poliklinik di Instalasi Rawat Jalan Rumah Sakit Islam Darus Syifa', antara lain adalah :

- Poliklinik Umum
- Poliklinik Gigi
- Poliklinik Anak
- Poliklinik Paru
- Poliklinik Mata
- Poliklinik Bedah Umum

- Poliklinik Kandungan
- Poliklinik Jantung
- Poliklinik Syaraf
- Poliklinik Rehab Medik

Untuk Instalasi Rawat Inap terdapat total 50 tempat tidur dengan perincian sebagai berikut :

Tabel 2.1 Jumlah Kamar RSI Darus Syifa' Benowo Surabaya

Kelas Kamar	Jumlah TT
V I P	1
I	1
II	6
III	42
Total	50

Rumah Sakit juga mempunyai 7 ruangan khusus untuk perawatan pasien rawat inap, antara lain :

- Ruang NICU (Neonatus Intensive Care Unit)
- Ruang ICU (Intensive Care Unit)
- Ruang Operasi
- Kamar Bersalin
- Ruang RR (Recovery Room)
- Ruang OK

Selain ruangan- ruangan diatas Rumah Sakit masih dilengkapi dengan Instalasi Penunjang Medis, antara lain :

- Laboratorium
- Radiologi

Untuk rawat darurat Rumah Sakit Islam Darus Syifa' siap dengan adanya UGD (Unit Gawat Darurat) yang siap 24 Jam.

Rumah Sakit Islam Darus Syifa' juga dilengkapi dengan Instalasi Farmasi yang menyediakan obat-obatan yang diperlukan oleh pasien rumah sakit, baik untuk rawat jalan maupun rawat inap.

Jumlah seluruh tenaga kerja di rumah sakit Islam Darus Sifa' adalah : 73 orang, sehingga dapat diketahui bahwa rasio tenaga medis per TT adalah 73:50 (1,46:1).

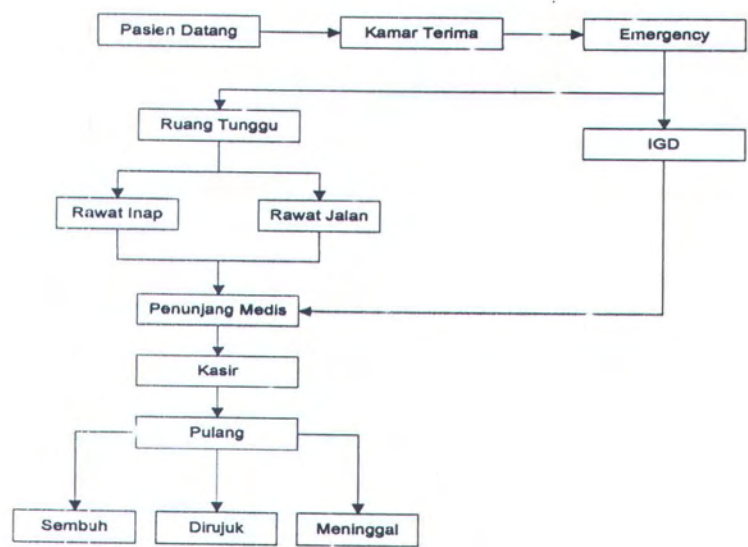
2.6.5 Existing Sistem di Rumah Sakit Islam Darus Syifa' Benowo Surabaya

Keadaan sistem informasi di Rumah Sakit Islam Darus Syifa' sekarang adalah sebagai berikut :

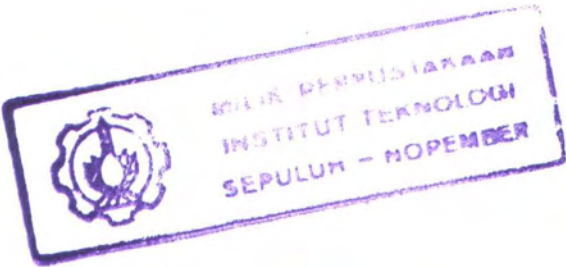
- Rumah Sakit Islam Darus Syifa' masih menggunakan berkas-berkas manual untuk menjalankan segala aktifitas-aktifitas dalam pelayanan terhadap pasien. Setelah selesai pelayanan, data manual tersebut baru dipindahkan ke komputer ke dalam program sekelas Word Processing atau Spreadsheet.
- Terdapat aplikasi khusus untuk pendaftaran pasien di Kamar Terima dan pembelian obat di Apotek. Namun masih belum online dengan jaringan yang ada. Adapun kelemahan-kelemahan dari aplikasi yang lama:

1. Aplikasi yang lama hanya mencatat proses pendaftaran pasien dan pembelian pasien.
 2. Laporan yang tersedia dalam aplikasi lama tidak lengkap, sehingga banyak informasi yang diperlukan tidak dapat disajikan.
 3. Software tidak terintegrasi secara menyeluruh ke seluruh bagian.
- Rumah Sakit Islam Darus Syifa' mempunyai 10 buah komputer untuk operasional dengan spesifikasi antara lain :
 - Processor : Intel Celeron 1 GHz dan Intel Celeron 733 Mhz
 - Memori SDRAM 128 MB di tiap-tiap komputer
 - Lancard di tiap-tiap komputer
 - Semua komputer sudah terhubung dengan jaringan dengan Topologi Star.
 - Mekanisme pengawasan dan kontrol yang ada dalam sistem berjalan dirasakan kurang, meliputi :
 1. pengawasan atas input administrasi penderita (rawat inap)
 2. pengawasan atas transaksi poliklinik
 3. pengawasan atas transaksi Apotek
 4. pengawasan atas pemasukan data rekam medik pasien, diagnosa, dan terapi dalam ruang perawatan.
 - Kesulitan dalam pengumpulan informasi yang dikehendaki
 Disebabkan karena masih manualnya sistem, maka sering ditemukan kesulitan untuk mendapatkan informasi yang dikehendaki, terutama informasi yang melibatkan pengumpulan data dari beberapa departemen.

- Alur pasien yang berlaku di rumah sakit saat ini :



Gambar 2.8 Alur Pasien Rumah Sakit

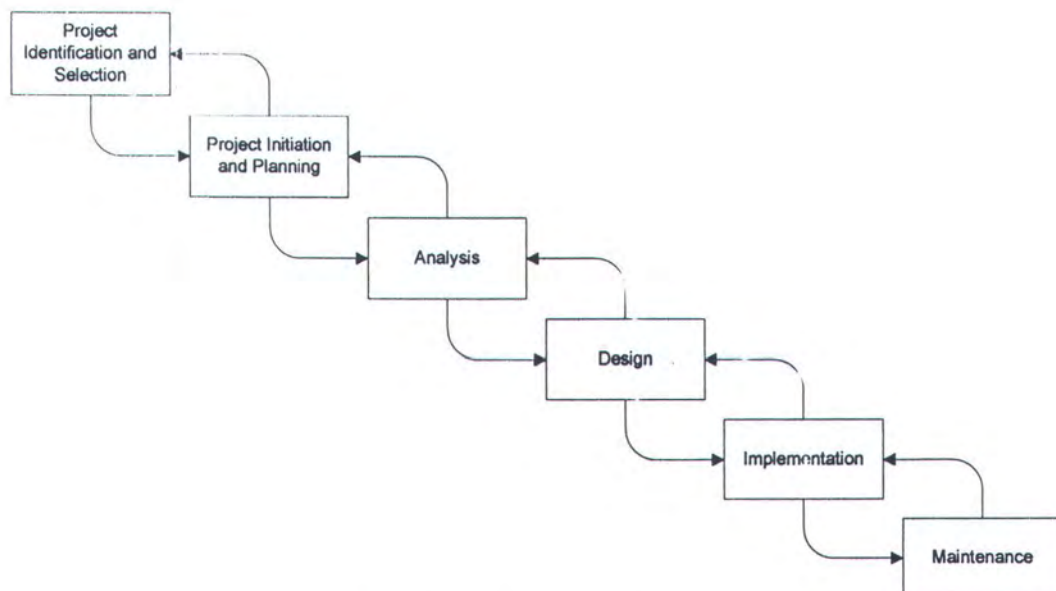


BAB III

KEBUTUHAN SISTEM

3.1 Metodologi Mencari Kebutuhan Sistem

Metodologi yang dipakai penulis dalam pembuatan perangkat lunak Sistem Informasi Pasien dan Rekam Medik mengacu pada System Development Life Cycle (SDLC) dalam buku “Modern System Analysis and Design” karangan Jeferey A. Hoffer, Joey F. Goerge dan Joseph S. Valacich, dengan menggunakan metode *Rapid Application Development (RAD)* dengan urutan seperti dibawah ini:



Gambar 3.1 Rapid Application Development

Dengan menggunakan metode RAD ini penulis dapat membuat perangkat lunak secara bertahap yang dimulai dari tahapan “Project Identification and Selection” sampai dengan tahap “Maintenance”. Keuntungan dengan menggunakan metode ini apabila ada kesalahan-kesalahan yang terjadi dalam

tahap sebelumnya, penulis dapat kembali untuk memperbaikinya dan kemudian dapat melanjutkan lagi ke tahapan selanjutnya sesuai dengan urutan tahap dalam metode RAD, sehingga penulis tidak perlu mengulang kembali dari tahapan awal lagi misal seperti dalam metode *Waterfall*.

Adapun tahapan-tahapan dalam proses RAD ini adalah sebagai mana berikut :

1. Project Identification and Selection

Penulis mengidentifikasi keinginan dari rumah sakit untuk membuat sistem informasi pasien yang baru.

2. Project Initiation and Planning

Penulis menginvestigasi masalah-masalah dalam sistem atau kesempatan-kesempatan yang ada dan memberikan penjelasan alasan apakah sistem informasi pasien perlu dibuat di rumah sakit atau tidak.

3. Analysis

Penulis mempelajari keseluruhan prosedur yang dipakai dalam rumah sakit. Dalam tahap ini dibagi dalam beberapa fase, antara lain :

1. Requirement Determination

Penulis menentukan apa yang diinginkan pihak rumah sakit pada sistem yang baru dengan mempelajari sistem yang sudah ada dan dokumen-dokumen yang ada, dengan melakukan survey dan wawancara.

2. Dari hasil Requirement Determination Penulis mengembangkannya ke dalam bentuk System Context Diagram dan Data Flow Diagram (DFD),

serta membuat Entity Relationship Diagram dalam bentuk Conceptual Data Model (CDM) dengan menggunakan aplikasi Power Designer 6.

4. Design

Dari hasil tahap Analysis diatas Penulis mendapatkan DDL (Data Definition Language) hasil dari peng-generate-an CDM dan menghasilkan Physical Tables, kemudian Penulis mendesain segala aspek dalam sistem mulai dari form input dan output dilayar, laporan-laporan yang dihasilkan, dan segala proses dalam sistem yang akan dibangun.

5. Implementaion

Penulis mengimplementasikan hasil dari design diatas ke dalam coding program serta melakukan uji coba aplikasi yang dihasilkan.

6. Maintenance

Penulis melakukan pengamatan jalannya program, dan memperbaiki apabila ada kerusakan-kerusakan.

3.2 Hasil-Hasil yang Didapatkan

Dari langkah-langkah diatas didapatkan hasil-hasil seperti dibawah ini :

1. Project Identification and Selection

Hasil dari tahap ini tercantun dalam Bab I dalam Latar Belakang yang menyebutkan bahwa Sistem Informasi Pasien dan Rekam Medik sangat dibutuhkan untuk mengorganisir segala aktifitas pasien rumah sakit dengan cepat, akurat, mudah dan efisien guna mendukung peningkatan pelayanan di Rumah Sakit Islam Darus Syifa'.

2. Project Initiation and Planning

Hasil dari tahap ini tercantum dalam Bab I dalam Permasalahan dan Tujuan pembuatan Sistem Informasi Pasien dan Rekam Medik dan Batasan-batasan dalam pembuatan Sistem Informasi Pasien dan Rekam Medik.

Adapun beberapa kelemahan-kelemahan dalam proses manajemen di rumah sakit saat ini, antara lain :

1. Bahwa sistem terdahulu masih menggunakan sistem manual dan belum seluruhnya terkomputerisasi dengan baik, sehingga menyulitkan pihak manajemen untuk mengolah data-data yang ada.
2. Adanya ketidakcocokan data antara data daftar pasien yang diperiksa di poliklinik dengan daftar pasien yang membayar karcis. Kesimpulannya masih ada pasien yang tidak membayar karcis tapi langsung pulang.
3. Dengan manualnya sistem, banyak terjadi data-data pasien yang tertukar tempat dengan data pasien lain.
4. Adanya ketidakcocokan antara tindakan dokter atau perawat dengan honor dokter atau perawat yang didapat.
5. Pasien mengeluhkan belum adanya faktur yang standar di rumah sakit, yang masih ditulis manual dengan tangan.

Juga dihasilkan kesepakatan mengenai batasan-batasan dalam pembuatan sistem informasi serta asumsi-asumsi yang ada, antara lain :

1. Sistem hanya melakukan pencatatan segala aktifitas-aktifitas pasien yang ada di rumah sakit.

2. Sistem tidak mencakup dengan sistem keuangan rumah sakit, sistem kepegawaian, sistem stock control di apotik, dan sistem aset manajemen rumah sakit.
 3. Diasumsikan bahwa pasien-pasien yang datang harus mendaftar di kamar terima terlebih dahulu.
 4. Aplikasi diharapkan dapat berjalan dengan komputer-komputer yang sudah ada di rumah sakit. Apabila ada tambahan pihak rumah sakit bersedia menambah jumlah komputer yang ada.
3. Analysis

Hasil dari tahap ini didapatkan setelah Penulis melakukan pengumpulan data, survey serta wawancara-wawancara dengan unit yang terkait di Rumah Sakit Islam Darus Syifa', antara lain :

1. Requirement Determination

Dalam tahap ini Penulis melakukan survey dan wawancara per bagian yang berhubungan langsung dengan pasien di rumah sakit. Adapun pihak-pihak yang diwawancarai, antara lain :

1. Direktur Rumah Sakit : dr. Achmad Hernowo W
2. Kepala Bagian Personalia Rumah Sakit : Akub Zeta S.Si
3. Apoteker Rumah Sakit Renny Tri Wahyuni, S.Si Apt
4. Beberapa staf rumah sakit.

Rangkuman pertanyaan-pertanyaan yang diajukan oleh Penulis adalah sebagaimana berikut :

1. Apakah tujuan dan latar belakang didirikannya rumah sakit ini?

2. Apakah sudah ada sistem informasi yang digunakan di rumah sakit?
3. Bagaimana alur pasien rumah sakit sekarang? Kelemahan dan keuntungan apakah yang didapatkan dari alur pasien sekarang?
4. Bagaimana koordinasi tiap departemen di rumah sakit dalam penanganan pasien saat ini?
5. Bagaimana sistem pentarifan pada pasien rumah sakit?
6. Bagaimana sistem penghargaan harga obat di apotek?
7. Bagaimana sistem pembayaran pasien ke rumah sakit pada saat ini?
8. Sistem apakah yang kemungkinan dikehendaki oleh pihak rumah sakit?
9. Proses apa saja yang dilalui pasien ketika berada di per departemen rumah sakit?
10. Kemudahan-kemudahan apa saja yang diinginkan dalam sistem yang baru?
11. Bagaimana bentuk Laporan-laporan yang dihasilkan oleh Rumah Sakit?

Hasil dari jawaban diatas dijabarkan oleh Penulis per unit-unit pelayanan dalam rumah sakit, seperti dibawah ini :

a. Kamar terima

Kamar teri na adalah tempat penerimaan pasien pertama kali. Disini akan dilakukan beberapa pencatatan untuk mengambil data-data pasien yang mendaftar.

Adapun aktifitas-aktifitas dalam Kamar Terima adalah sebagai berikut:

- Di kamar terima akan dicatat segala data mengenai pasien rawat jalan, rawat inap, serta gawat darurat yang datang, antara lain: Nama Lengkap, Alamat Lengkap, Tanggal Lahir, Jenis Kelamin, Golongan Darah, Status Pernikahan, Nama dan alamat Penanggung Jawab Pasien, Dari mana pasien datang, Kondisi Masuk, Diagnosa Masuk, Dokter yang merawat dan lain-lain.
- Di kamar terima pasien akan dibedakan berdasarkan tipe pasien yang menentukan pembayaran perawatan nantinya.
- Untuk Rawat Jalan akan dicatat poliklinik tujuan yang juga akan menentukan besarnya karcis yang akan dibayar.
- Untuk Rawat Inap akan dicatat ruangan perawatan serta kamar yang akan ditempati dan kelas perawatan.

Hasil survey:

1. Proses pencarian data Pasien lama, yang disebabkan
 - a. Pencarian informasi / data pasien hanya berdasarkan initial awal, jadi apabila banyak nama pasien yang kembar maka pencarian pasien akan lama.
 - b. Banyak terjadi nomor rekam medik yang kembar karena belum otomatisnya sistem dalam pembuatan nomor rekam medik.
2. Karcis masih menggunakan karcis cetakan biasa.
3. Untuk menghitung pasien yang datang pada hari ini masih manual.

b. Rekam Medis

Bertugas untuk mengumpulkan data dari masing-masing ruang perawatan dan unit penunjang.

Hasil survey:

1. Proses pengumpulan data dari masing-masing ruang perawatan dan unit penunjang dilakukan secara manual.
2. Proses perhitungan dan rekapitulasi data dilakukan secara manual.

c. Kamar Perawatan (Rawat Jalan, UCD, dan Rawat Inap)

Setelah Pasien diperiksa oleh Dokter atau Perawat maka staff Perawat wajib melakukan pengisian beberapa form, antara lain :

- Di dalam kamar perawatan baik rawat jalan, rawat darurat, Gawat Darurat, serta penunjang medis akan dilakukan pencatatan Rekam Medik Pasien dalam bentuk SOAP (Subjective, Objective, Assesment, Planning) yang penjelasannya adalah sebagai berikut:
 1. Subjective : keluhan pasien ketika datang
 2. Objective : hasil pemeriksaan
Meliputi juga hasil laboratorium dan radiologi.
 3. Assesment : diagnosa pasien
 4. Planning : rencana selanjutnya terhadap pasien
Meliputi juga terapi pasien serta resep dokter yang diberikan.
- Di dalam Kamar Perawatan juga dicatat segala tindakan yang dilakukan oleh dokter atau perawat, untuk kemudian ditagihkan biayanya di kasir.
- Dilakukan pencatatan diagnosa dan terapi pasien.

Hasil survey:

1. Daftar harga tindakan masih bisa dirubah (seharusnya hanya operator yang berwenang yang berhak mengubah harga).
2. Khusus di Laboratorium belum adanya layout pemeriksaan yang bisa langsung dimasukkan, sehingga untuk menulis hasil pemeriksaan menjadi lama.
3. Penghitungan tindakan-tindakan yang sudah dilakukan oleh pasien rawat inap masih manual.

d. Apotik

Apabila pasien mendapat resep obat, pasien akan menebusnya melalui apotik. Apotik akan mencatat segala transaksi obat pasien.

- Pada pasien rawat jalan, pasien luar serta pasien gawat darurat biaya obat langsung dibayarkan di apotik. Setelah membayar pasien akan mendapatkan kuitansi atau faktur pembelian obat. Dan apabila obat yang disediakan tidak ada maka pasien akan mendapatkan kopi resep.
- Untuk rawat inap biaya obat akan ditagihkan ketika pasien pulang.

Hasil survey:

1. Daftar harga obat masih bisa dirubah (seharusnya hanya operator yang berwenang yang berhak mengubah harga).
2. Untuk pembelian oleh berbagai tipe pasien, apotik mengalami kesulitan dalam menentukan harga jual, sehingga proses penjualan obat menjadi lama.

Untuk hasil-hasil kebutuhan sistem dapat dilihat pada sub bab terakhir pada Bab III ini.

2. Untuk pembuatan System Context Diagram, Data Flow Diagram (DFD) dan Conceptual Data Model (CDM) akan dijelaskan lebih lanjut di dalam Bab IV.

2. Design

Hasil dari tahap ini akan dibahas lebih lanjut di dalam Bab IV.

3. Implementation

Hasil dari tahap ini akan dibahas lebih lanjut di dalam Bab V.

4. Maintenance

Penulis tidak melakukan tahap Maintenance, Penulis hanya melakukan Uji Coba Sistem yang akan dibahas lebih lanjut di dalam Bab V.

3.3 Ringkasan Kebutuhan Sistem

Dari hasil-hasil yang didapatkan di atas, dapat disimpulkan kebutuhan-kebutuhan yang diperlukan oleh sistem, antara lain :

- Sistem mampu mengintegrasikan unit-unit pelayanan di rumah sakit ke dalam satu sistem.

Bahwa aplikasi yang dihasilkan tidak terpisah-pisah per bagian rumah sakit (misal : aplikasi pendaftaran dan aplikasi apotek), melainkan terintegrasi menjadi satu, sehingga memudahkan pihak manajemen untuk mengumpulkan data dan mengolahnya lagi.

- Sistem bisa mengelola user yang menjalankan aplikasi dengan membatasinya sesuai dengan kewenangannya masing-masing user.

Setiap user yang menjalankan program aplikasi ini mempunyai otorisasi yang berbeda-beda. Misal user Amir yang bertugas di Kasir hanya bisa menjalankan form-form dan laporan-laporan yang hanya berhubungan dengan pembayaran-pembayaran pasien seperti pembayaran pulang rawat inap dan pembayaran hutang pasien, sedangkan Budi yang bertugas di kamar perawatan hanya bisa menjalankan form-form dan laporan-laporan yang berhubungan dengan perawatan pasien, seperti rekam medik dan diagnosa pasien.

- Sistem mampu melakukan pencatatan registrasi pasien rawat jalan, rawat inap dan instalasi gawat darurat.

Sistem dapat mencatat ketiga tipe pendaftaran pasien dan mengklasifikasikan sesuai dengan kedatangannya. Dimana dalam rawat jalan dan gawat darurat mempunyai sistem yang hampir sama, namun kedatangan mereka dicatat berbeda.

- Sistem bisa membedakan tipe pasien yang melakukan pendaftaran.

Terdapat 4 tipe pasien yang ada di rumah sakit, antara lain :

1. Pasien Umum : Pasien yang datang langsung.
2. Pasien ASKES : Pasien mempunyai kartu anggota ASKES,
3. Pasien ASTEK : Pasien mempunyai kartu anggota JAMSOSTEK,
4. Pasien Rumah Sakit : Pasien tercatat sebagai pegawai rumah sakit atau kerabatnya.

5. Pasien Perusahaan : Pasien dari perusahaan yang bekerja sama dengan rumah sakit.

- Sistem bisa mencetak karcis atau kuitansi bukti pendaftaran pasien secara otomatis.

Ketika melakukan pendaftaran sistem dapat otomatis atau manual mencetak karcis atau kuitansi untuk diberikan pada pasien.

- Sistem bisa membuat nomor rekam medik pasien secara otomatis.

Nomor rekam medik pasien di Rumah Sakit Islam Darus Syifa' mempunyai format 6 digit, dimana penomoran berdasarkan kedatangan pasien baru. Dimana dengan otomatisnya nomor rekam medik, nomor rekam medik ganda atau sama tidak akan terjadi lagi.

- Sistem mampu menampung keterangan-keterangan pasien sesuai dengan format data rekam medik rumah sakit.

Adapun keterangan-keterangan pasien yang umum dicatat adalah :

nama, alamat lengkap dengan RT, RW, kecamatan, kelurahan dan kota, tempat dan tanggal lahir, golongan darah, status perkawinan, keterangan pekerjaan, nama dan alamat penanggung jawab, nama dan alamat kerabat dekat, dan keterangan tentang pasien.

- Sistem bisa mencatat segala tindakan medis dan non medis rumah sakit.

Tindakan-tindakan terhadap pasien di rumah sakit mempunyai kategori yang berbeda-beda, dimana sistem dituntut dapat menyesuaikan diri. Kategori-kategori tersebut untuk membedakan tindakan-tindakan yang diberikan ke

pasien dan nantinya untuk pasien rawat inap dilaporkan sebagai rincian tindakan yang didapatkan ketika dirawat di rumah sakit.

- Sistem bisa mencatat tindakan-tindakan pasien di ruangan perawatan.

Segala tindakan terhadap pasien baik yang dilakukan oleh perawat ataupun dokter dapat tercatat dalam sistem.

- Sistem mampu mencatat rekam medik pasien dalam format SOAP (Subjective, Objective, Assesment, Planning).

Untuk memudahkan dalam pencatatan rekam medik digunakan sistem SOAP, yang membedakan-membedakannya ke dalam Subjective, Objective, Assesment, dan Planning.

- Sistem bisa membuat layout dari pemeriksaan laboratorium sesuai dengan format pemeriksaan di rumah sakit.

Untuk memudahkan pencatatan hasil pemeriksaan di laboratorium dibutuhkan suatu layout yang dapat menampung macam-macam pemeriksaan di laboratorium berikut dengan item-item per pemeriksaannya. Misal pemeriksaan Darah Lengkap terdapat item pemeriksaan Hemoglobin, Lekosit, LED dan lain-lain.

- Sistem bisa mencetak Hasil Pemeriksaan Laboratorium.

Dari hasil pemeriksaan laboratorium, hasilnya dapat dimasukkan melalui layout-layout pemeriksaan diatas, dan kemudian dapat dicetak untuk diserahkan ke pasien sebagai hasil pemeriksaan resmi rumah sakit.

- Sistem mampu mencatat diagnosa dan terapi yang dilakukan ke pasien.

Setiap kedatangan di rumah sakit baik rawat jalan ataupun rawat inap, rumah sakit selalu mencatat diagnosa sakit pasien yang mempunyai jenis sebagaimana berikut, antara lain :

1. Diagnosa Sementara
2. Diagnosa Masuk Rumah Sakit (Dx In)
3. Diagnosa Keluar Rumah Sakit (Dx Out)
4. Diagnosa Sebelum Operasi (Dx PreOp)
5. Diagnosa Setelah Operasi (Dx PostOp)
6. Diagnosa Patologi Anatomi (Dx PA)

Untuk rawat jalan hanya dicatat Diagnosa Sementara ketika pasien datang, dan dicatat Diagnosa Keluar Rumah sakit ketika pasien pulang serta dicatat juga apakah diagnosa ini baru bagi pasien atau diagnosa penyakit lama yang pernah diderita oleh pasien.

Sedang untuk rawat inap dicatat segala diagnosa diatas sesuai dengan keadaan pasien di rumah sakit.

- Sistem bisa mencatat segala pembayaran pasien sesuai dengan tipe pasien baik tunai maupun piutang.

Adapun sistem pembayaran pasien disesuaikan dengan tipe pasien ketika pasien mendaftar, dimana sistem harus bisa mencatat segala pembayaran pasien per tipe pasien.

- Sistem bisa mencatat transaksi hutang pasien.

Apabila pasien berhutang pada rumah sakit, sistem dapat mencatatnya ke dalam daftar hutang, yang dapat ditagihkan ke pasien pada hari yang telah ditentukan.

- Sistem bisa menjalankan proses pembayaran hutang pasien yang belum terbayar.

Sistem dapat mengelola daftar hutang pasien, dimana apabila pasien membayar secara otomatis pasien dihapuskan dari daftar hutang pasien.

- Sistem bisa mengeluarkan rincian pembayaran pasien rawat inap.

Ketika pulang atau ketika diminta oleh pasien, sistem dapat mengeluarkan rincian biaya rumah sakit yang sesuai dengan kategori-kategori tindakan dalam jangka waktu dari pasien datang sampai pasien meminta rincian tersebut.

- Sistem bisa mencetak rincian pembayaran pasien rawat inap.

Sistem dapat mencetak rincian biaya rumah sakit diatas, sebagai bukti pembayaran terakhir pasien rawat inap.

- Sistem bisa mencatat segala pembelian obat pasien, baik pasien luar maupun pasien rumah sakit.

Terdapat 3 tipe pasien yang membeli obat di apotek, antara lain :

1. Pasien Rawat Jalan

Untuk pasien rawat jalan, dicatat nomor rekam medik dari pasien, obat-obat yang dibeli, dan pembayaran pasien baik hutang atau tunai.

2. Pasien Rawat Inap

Untuk pasien rawat inap, hanya dicatat nomor rekam medik dari pasien dan obat yang dibeli saja. Untuk pembayarannya ditagihkan ketika pasien pulang.

3. Pasien Luar Rumah Sakit

Untuk pasien luar rumah sakit, dicatat nama, alamat, obat-obat yang dibeli, serta pembayaran pasien selalu tunai.

- Sistem bisa membedakan harga jual ke pasien sesuai dengan jenis pembelian pasien dan tipe pasien.

Terdapat macam-macam pembelian obat oleh pasien, antara lain:

1. Pembelian Bebas
 2. Pembelian Umum
 3. Pembelian Perusahaan
 4. Pembelian Discount 15%
 5. Pembelian Discount 20%
 6. Pembelian Discount 25%
 7. Pembelian Karyawan 15%
 8. Pembelian Karyawan 20%
 9. Pembelian Karyawan 25%
- Sistem mampu mencatat keterangan-keterangan obat sesuai dengan format keterangan obat di rumah sakit sekarang.

Adapun keterangan-keterangan obat yang harus dicatat antara lain nama obat, pabrik, jenis obat, bahan-bahan yang terkandung di dalamnya, harga beli,

satuan beli, harga pokok pembelian, harga netto apotek, harga jual, satuan jual dan konversi satuan beli ke jual.

- Sistem mampu membedakan kategori dari obat yang ada.

Terdapat beberapa kategori obat, antara lain obat Narkotika, Psikotropika, Bebas, Generik, dan lain-lain. Dimana sistem dapat mencatat segala penjualan obat sesuai dengan kategorinya masing-masing.

- Sistem mampu mengelola harga R per obat yang dijual.

Harga R adalah harga tambahan yang dibebankan pada harga jual obat yang merupakan komisi untuk staff apotek. Terdapat 2 tipe R, yaitu untuk obat non racikan, dan obat racikan.

- Sistem bisa mencetak faktur dan kuitansi penjualan obat.

Obat-obat yang dimasukkan dalam transaksi penjualan dalam sistem harus dapat dicetak dalam bentuk faktur dan kuitansi sebagai bukti pembelian obat oleh pasien rumah sakit.

- Sistem bisa melakukan pencatatan pemakaian kamar rawat inap.

Ketika pasien mendaftar rawat inap, secara otomatis pasien harus menyewa kamar untuk perawatannya di rumah sakit. Ketika dalam perawatan rawat inap, pasien dapat berpindah-pindah kamar sesuai keinginannya. Sistem dituntut untuk dapat melakukan segala pencatatan aktifitas pasien ketika pasien masuk kamar, pindah kamar dan keluar dari kamar perawatan. Dimana yang dicatat adalah No rekam Medik, kamar yang ditempati, tanggal dan jam masuk, tanggal dan jam keluar, dan lama perawatan.

- Sistem bisa menghasilkan perhitungan statistik pemakaian kamar rawat inap (BOR,BTO,ALOS).

Sistem dapat menghasilkan perhitungan Bed Occupancy Ratio (BOR), Bed Turnover, Average Length of Stay (ALOS) yang dihasilkan dari data pemakaian kamar diatas.

- Sistem mampu menghasilkan laporan-laporan untuk rumah sakit, antara lain :

1. Laporan Register Pendaftaran Harian Pasien Rawat Jalan dan IGD.
2. Laporan Register Pelayanan Harian Pasien Rawat Jalan dan IGD.
3. Laporan Sensus Harian Pasien Rawat Inap.
4. Laporan Morbiditas Pasien Rawat Jalan.
5. Laporan Morbiditas Pasien Rawat Jalan Surveilans.
6. Laporan Morbiditas Pasien Rawat Inap.
7. Laporan Morbiditas Pasien Rawat Inap Surveilans.
8. Laporan Ranging Diagnosa Pasien Rawat Jalan
9. Laporan Statistik Rumah Sakit
10. Laporan Rekam Medik Pasien.
11. Laporan Daftar Harga Tindakan.
12. Laporan Pemakaian Kamar Rawat Inap.
13. Laporan Daftar Harga Obat.
14. Laporan Daftar Penjualan Obat Per shift.
15. Laporan Pendapatan Karcis Rawat Jalan.
16. Laporan Daftar Hutang Pasien.
17. Laporan Pemeriksaan Dokter Harian Rawat Jalan.

BAB IV

ANALISA DAN DESAIN SISTEM

Pada bab ini Penulis membahas proses pembuatan perangkat lunak sistem informasi pasien dan rekam medik guna untuk memenuhi kebutuhan-kebutuhan sistem diatas. Pembuatan perangkat lunak ini dilakukan dalam beberapa tahapan.

4.1 Analisa Sistem

Hasil dari Requirement Determination dan Kebutuhan Sistem diatas dikembangkan ke dalam bentuk Proses Bisnis. Dari proses-proses dan entity-entity yang dihasilkan akan dikembangkan lagi ke bentuk Data Flow Diagram (DFD) dan Conceptual Data Model (CDM), dengan langkah-langkah sebagai mana berikut :

4.1.1 Proses Bisnis

Proses-proses yang terlibat dalam proses ini mempunyai beberapa entitas, antara lain :

1. Pasien

Entitas Pasien adalah entitas utama dalam proses-proses bisnis dalam sistem informasi pasien dan rekam medik.

2. Dokter



Entitas Dokter hanya bertugas menganalisa data pasien yang ada, kemudian melakukan pemeriksaan dan memberikan advise rekam medik kepada pasien.

3. Manajemen

Entitas Manajemen hanya menerima laporan dari proses bisnis yang ada.

Adapun proses-proses dalam pembuatan Sistem Informasi Pasien dan Rekam Medik adalah sebagai berikut :

1. Proses Rawat Jalan.

Adalah proses ketika pasien berada di pelayanan rawat jalan rumah sakit.

Proses Rawat Jalan pasien tersebut adalah sebagai berikut

- Proses Pendaftaran Rawat Jalan dan IGD :
 1. Pasien atau yang mewakili melakukan pendaftaran dengan menunjukkan kartu rumah sakit atau menyebutkan nomor id bagi pasien lama.
 2. Untuk pasien baru cukup menyebutkan identitasnya kepada petugas untuk kemudian diberikan kartu rumah sakit yang baru dengan nomor id yang unique.
 3. Pasien kemudian menyebutkan poli yang dituju atau menyebutkan gejala-gejala penyakitnya atau memberikan rujukan dari dokter kepada petugas. Petugas akan memilihkan poli yang dituju. Apabila pasien mempunyai tipe pasien khusus seperti pasien ASKES atau keluarga rumah sakit, pasien menyebutkan identitas

khususnya kepada petugas, dengan membawa bukti misal : pasien ASKES harus membawa kartu ASKES.

4. Petugas akan memasukkan data pendaftaran pasien ke dalam komputer sesuai dengan poli dan tipe pasien.

Adapun perintah SQL untuk memasukkan data pasien rawat jalan ke dalam database adalah sebagaimana berikut :

5. Kemudian pasien menuju kasir untuk membayar biaya karcis.
 6. Untuk pasien IGD pasien tidak perlu membayar karcis tapi langsung masuk ke ruangan untuk di periksa. Pembayaran karcis untuk pasien IGD dilakukan ketika pasien pulang.
 7. Pasien menuju ke poliklinik.
- Proses Tindakan dan Rekam Medik Rawat Jalan dan IGD :
 1. Petugas akan memanggil pasien sesuai dengan data pasien yang terdapat di komputer.
 2. Pasien menunjukkan bukti karcis kepada petugas.
 3. Pasien akan diperiksa oleh dokter dan hasil dari pemeriksaan akan dimasukkan ke dalam komputer dalam bentuk Subjective Objective, Assesment, Planning (SOAP) sebagai rekam medik pasien. Dicatat juga diagnosa dari pasien, terapi yang dilakukan, dan keterangan-keterangan dari pasien.
 4. Apabila ada tindakan khusus yang dilakukan dokter, pasien akan diberikan tanda bukti untuk melakukan pembayaran di kasir.
 - Proses Pengolahan Data Rawat Jalan :

Data-data dari proses-proses rawat jalan diatas akan diolah dan dikirimkan ke proses pembuatan laporan untuk dihasilkan laporan-laporan rawat jalan.

2. Proses Rawat Inap

Adalah proses ketika pasien berada di pelayanan rawat inap rumah sakit.

Proses Rawat Inap pasien tersebut adalah sebagai berikut

- Proses Pendaftaran Rawat Inap :
 1. Pasien atau yang mewakili melakukan pendaftaran dengan menunjukkan kartu rumah sakit bagi pasien lama, sedangkan untuk pasien baru cukup menyebutkan identitasnya kepada petugas untuk diberikan kartu rumah sakit yang baru dengan nomor id yang unique.
 2. Untuk rawat inap Pasien memberikan keterangan rujukan untuk rawat inap.
 3. Pasien memilih kamar perawatan dan kelas perawatan. Dan seperti pada pasien rawat jalan, apabila pasien mempunyai tipe yang khusus, pasien wajib menunjukkan bukti.
 4. Petugas akan memasukkan data pendaftaran pasien ke dalam komputer.
 5. Kemudian Pasien atau yang mewakili menuju kasir untuk membayar uang muka pendaftaran.
 6. Pasien menuju ke kamar perawatan.
- Proses Tindakan dan Rekam Medik Rawat Inap :

1. Di kamar perawatan pasien akan mendapat perawatan dari dokter atau perawat, dimana perawat akan mencatat segala perkembangan pasien ke dalam komputer dalam bentuk Subjective Objective, Assesment, Planning (SOAP) sebagai rekam medik pasien.
2. Perawat juga akan mencatat tindakan-tindakan yang dilakukan oleh dokter setiap hari, yang akan ditagihkan pada pasien ketika pasien pulang. Akan dicatat juga diagnosa dari pasien, terapi yang dilakukan, dan keterangan-keterangan dari pasien.

- **Proses Pengolahan Data Rawat Inap :**

Data-data dari proses-proses rawat inap diatas akan diolah dan dikirimkan ke proses pembuatan laporan untuk dihasilkan laporan-laporan rawat inap.

3. Proses Apotik.

Adalah proses ketika pasien melakukan pembelian obat atau alat kesehatan di apotik.

Proses Apotik tersebut adalah sebagai berikut :

- **Proses Pembelian Obat Umum :**

Pembelian oleh pasien yang tidak terdaftar di rumah sakit. Pasien datang membawa resep dari dokter atau membeli obat secara langsung. Petugas akan mencatat nama dan alamat dari pembeli serta nama dokter yang memberi resep. Kemudian petugas akan memasukkan

data-data transaksi ke dalam komputer dan mencetak faktur pembelian obat yang diberikan kepada pasien.

- **Proses Pembelian Obat Pasien Rumah Sakit :**

Pembelian oleh pasien yang terdaftar di rumah sakit. Pasien datang membawa resep dari dokter atau membeli obat secara langsung. Pasien menunjukkan ID Pasien kepada petugas. Petugas akan mencari data-datanya dan memasukkan data-data transaksi ke dalam komputer dan mencetak faktur pembelian obat yang diberikan kepada pasien.

- **Proses Pembelian Obat Rawat Inap :**

Pembelian oleh pasien rawat inap rumah sakit. Perwakilan dari pasien atau perawat yang merawat pasien akan memberikan resep dari dokter yang merawat pasien kepada apotik. Petugas akan mencari data-datanya dan memasukkan data-data transaksi ke dalam komputer. Transaksi obat untuk rawat inap akan dikirim ke proses pembayaran rawat inap untuk ditagihkan ketika pasien pulang dari rumah sakit.

- **Proses Pengolahan Data Apotik :**

Data-data dari proses-proses apotik diatas akan diolah dan dikirimkan ke proses pembuatan laporan untuk dihasilkan laporan-laporan apotik.

4. Proses Pembayaran Rumah Sakit Pasien.

Adalah proses ketika pasien melakukan pembayaran hutang rumah sakit.

Proses Hutang Pasien tersebut adalah sebagai berikut :

- **Proses Pembayaran Rawat Jalan dan IGD :**

1. Pasien membayar biaya karcis masuk sesuai dengan poliklinik dan akan mendapat tanda bukti karcis dari petugas.
 2. Pasien menunjukkan tanda bukti transaksi, petugas kemudian akan memasukkan transaksi ke dalam komputer dan mencetak faktur tindakan yang diberikan kepada pasien.
- Proses Pembayaran Rawat Inap :
 1. Pasien membayar uang muka pendaftaran rawat inap, kemudian petugas akan mencetak bukti pembayaran uang muka untuk diberikan kepada pasien.
 2. Pasien melakukan pembayaran akhir rawat inap, dimana petugas akan menunjukkan rincian biaya yang dibebankan kepada pasien selama pasien berada di rumah sakit, yaitu tindakan-tindakan dokter atau perawat, sewa kamar serta pemakaian obat dan alat kesehatan. Petugas kemudian akan memasukkan transaksi pembayaran ke dalam komputer dan mencetak faktur rawat inap yang diberikan kepada pasien.
 - Proses Pembayaran Obat :

Pasien Umum atau Pasien Rumah Sakit diatas membayar pembelian obat sesuai dengan jumlah pembelian yang dilakukan. Petugas akan memasukkan data pembayaran apotik dan mencetak faktur atau kuitansi pembelian obat untuk diberikan kepada pasien.

5. Proses Hutang Pasien.

Adalah proses ketika pasien melakukan pembayaran hutang rumah sakit.

Proses Hutang Pasien tersebut adalah sebagai berikut :

- Proses Pelayanan Pembayaran Hutang :

Pasien menunjukkan bukti hutang atau id pasien kepada petugas. Petugas akan mencari data hutang pasien di komputer dan akan menunjukkan rincian hutang kepada pasien.

- Proses Pembayaran Hutang Pasien :

Pasien membayar jumlah hutang sesuai dengan rincian hutang diatas. Petugas akan mencetak bukti pembayaran hutang pasien.

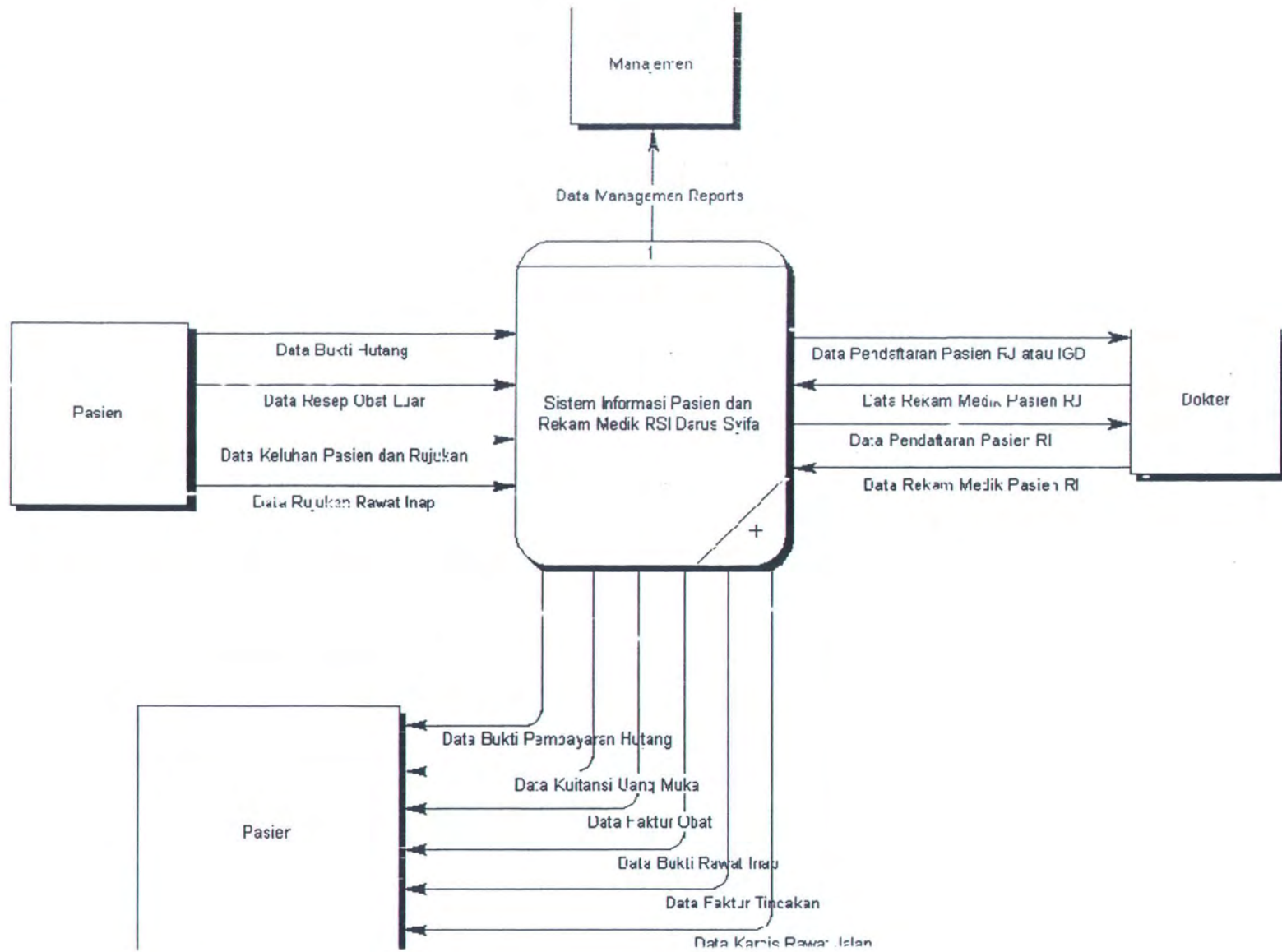
- Proses Pengolahan Data Hutang :

Data-data dari proses-proses hutang diatas akan diolah dan dikirimkan ke proses pembuatan laporan untuk dihasilkan laporan-laporan data hutang.

6. Proses Pelaporan.

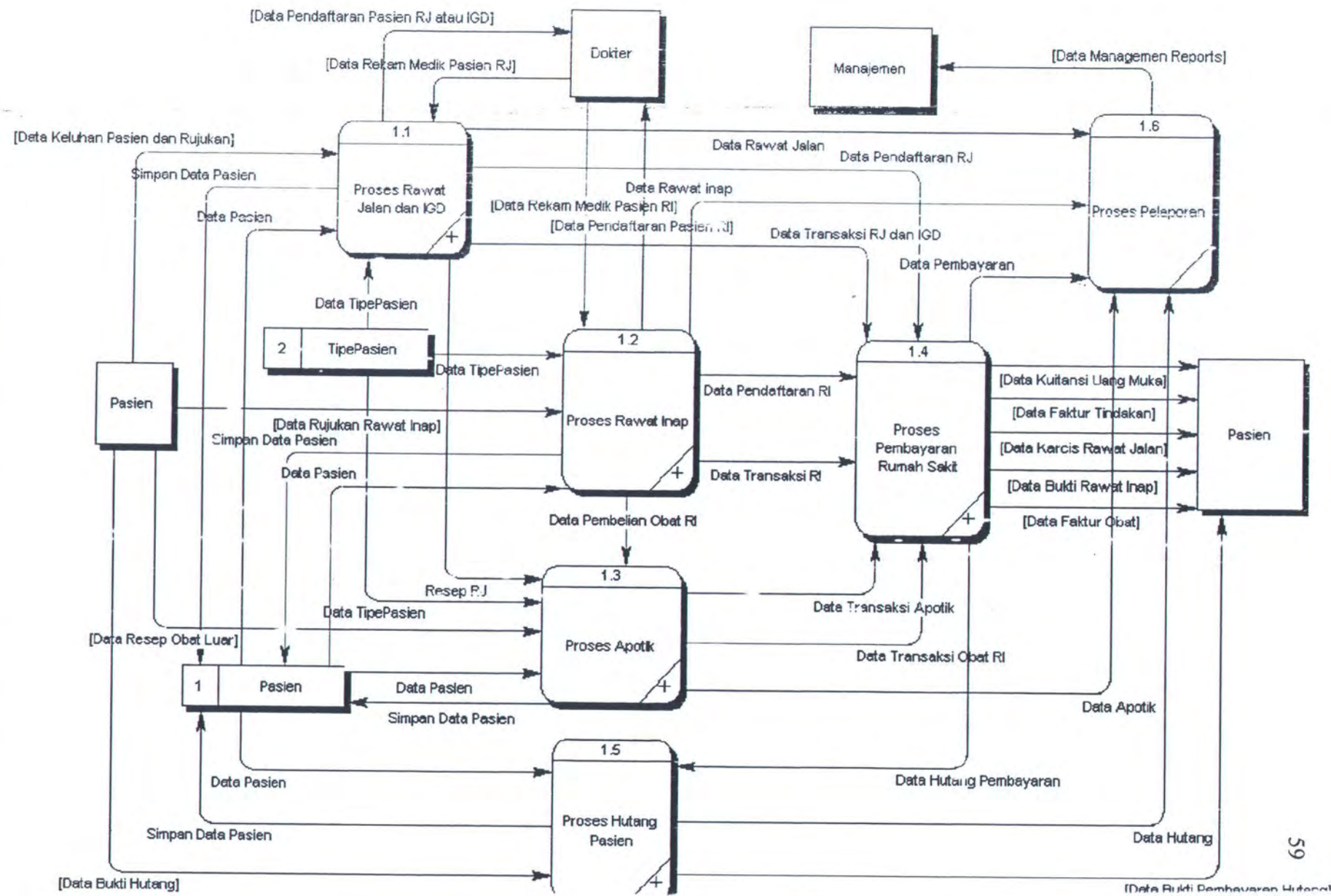
Data-data serta view-view yang dihasilkan dari proses-proses diatas akan diproses dan diolah sedemikian rupa sehingga menghasilkan laporan-laporan yang diperlukan pihak manajemen rumah sakit serta untuk pelaporan ke dinas kesehatan.

Gambaran dari proses-proses bisnis dalam bentuk System Context Diagram, DFD Level 0 serta hasil dari Decompose dari proses-proses diatas terdapat di halaman berikutnya.

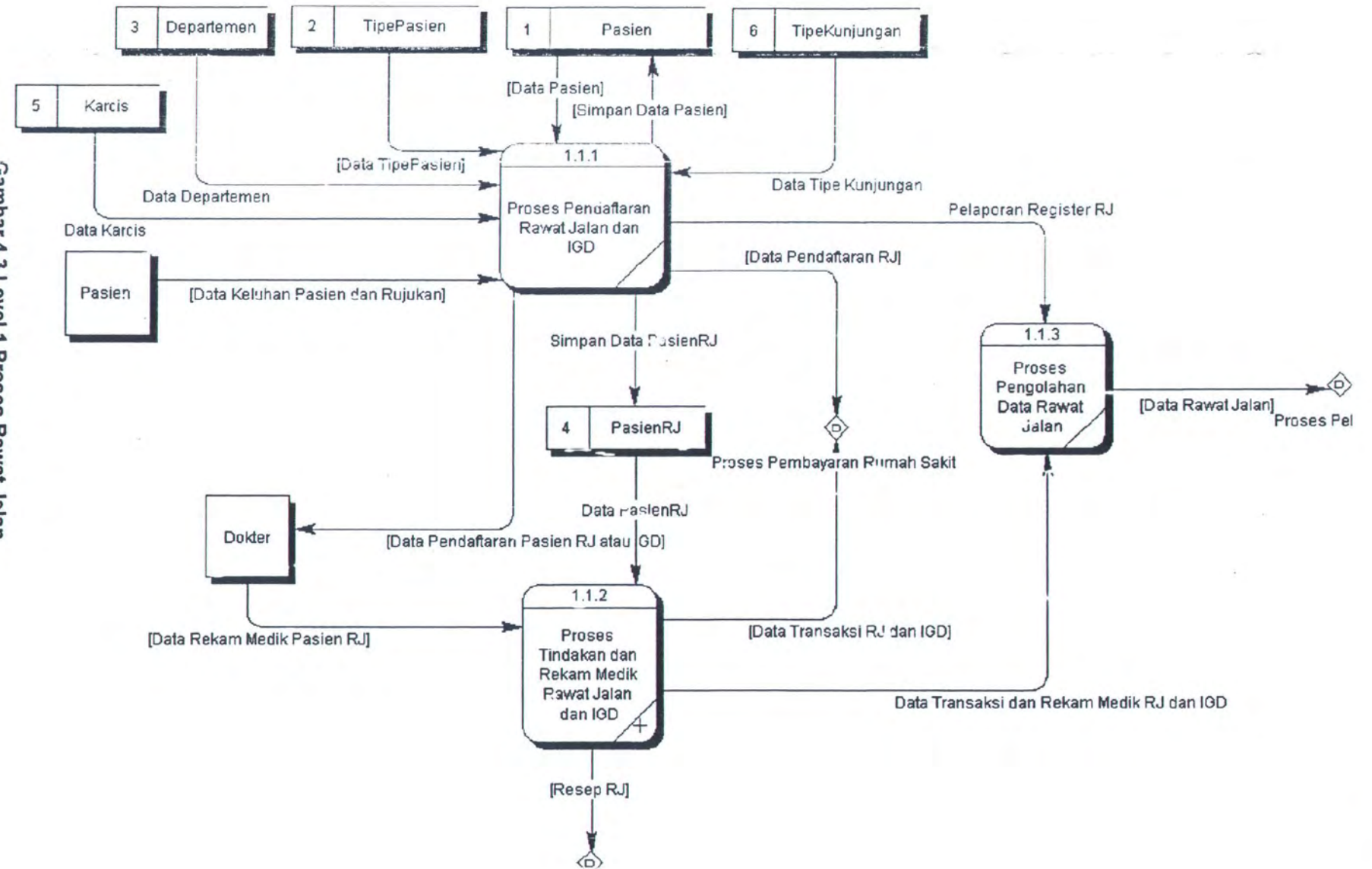


Gambar 4.1 System Context Diagram Sistem Informasi Pasien dan Rekam Medik RSI Darus Syifa Benowo Surabaya

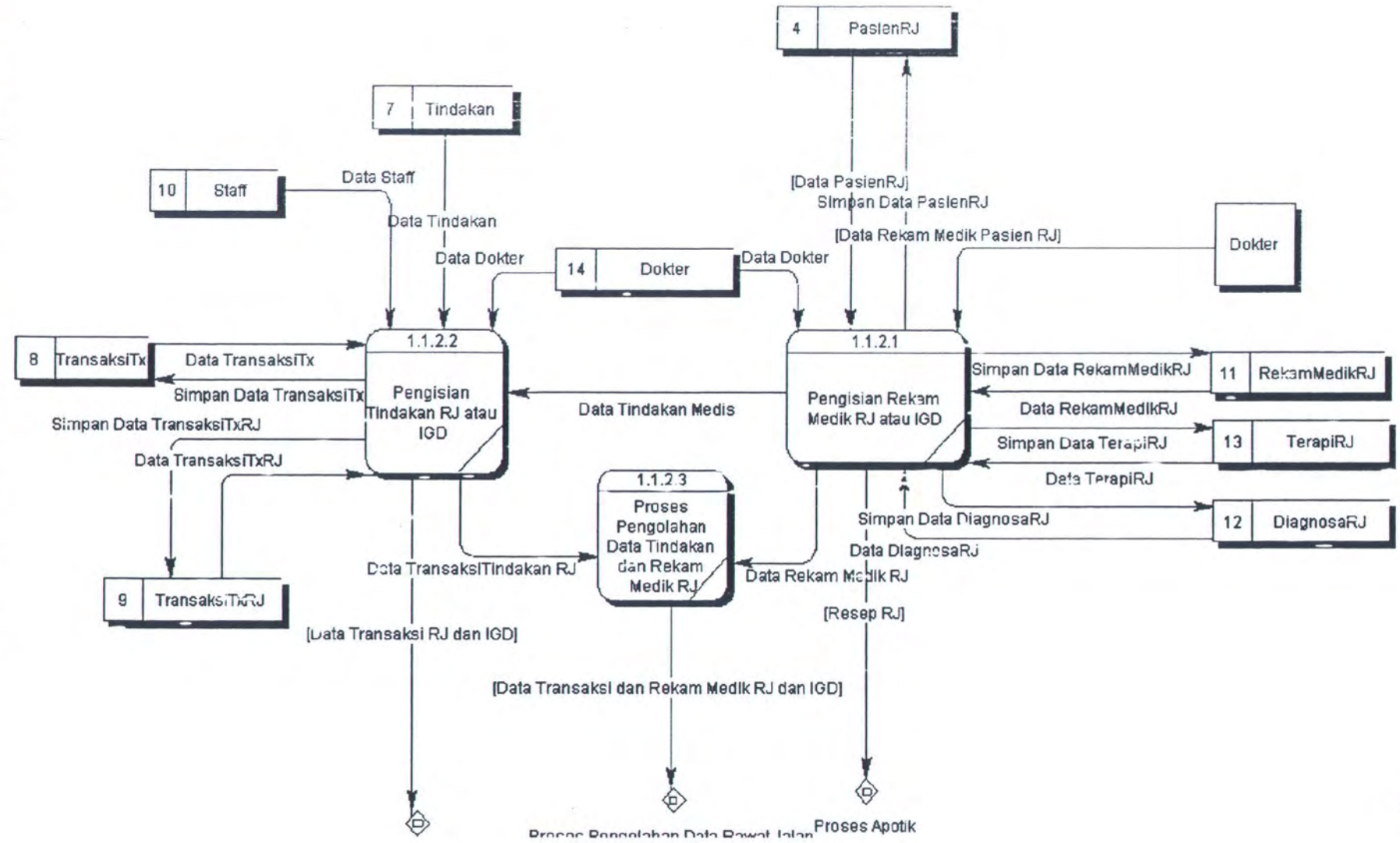
Gambar 4.2 Level 0 DFD Sistem Informasi Pasien dan Rekam Medik RSI Darus Syifa' Benowo Surabaya



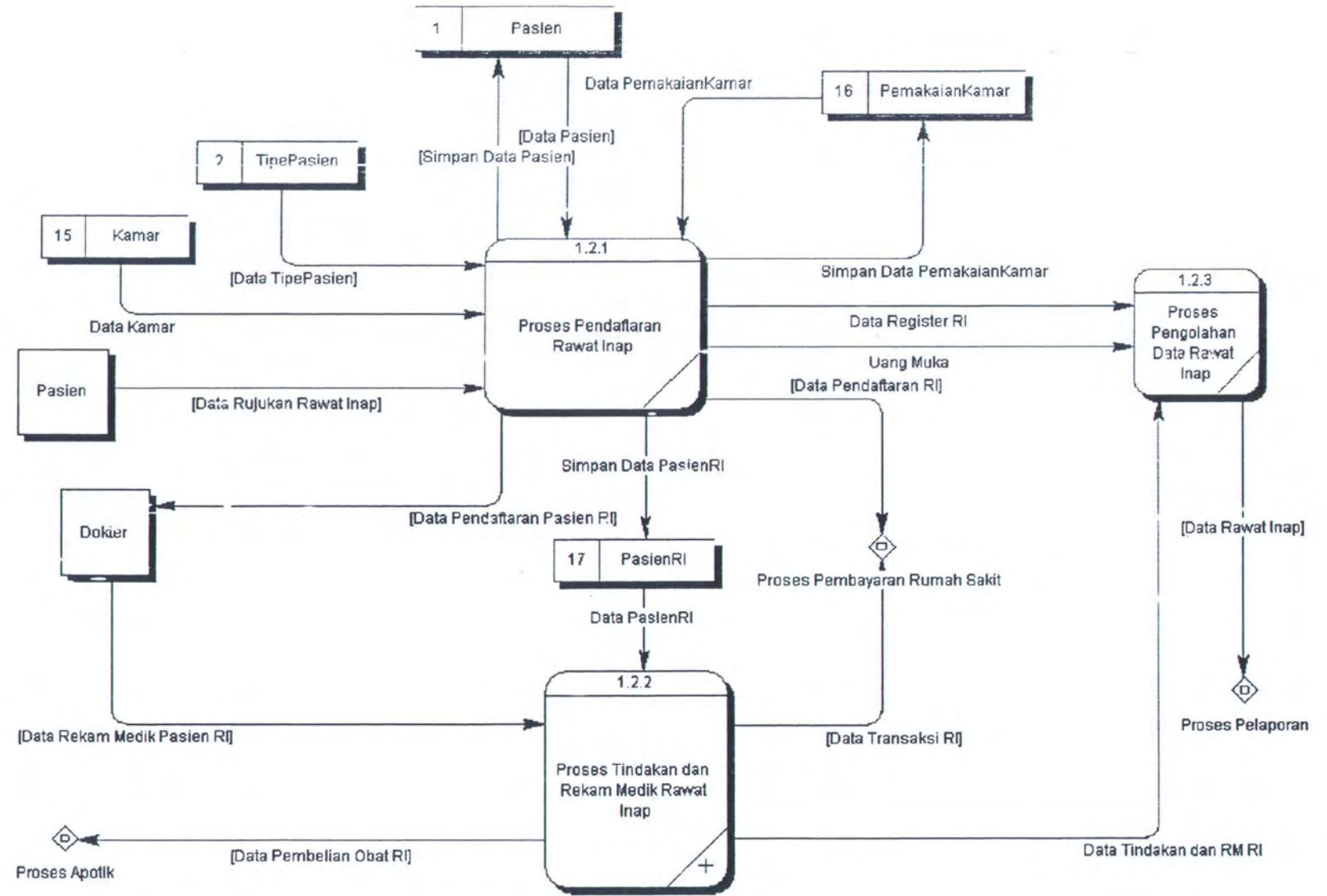
Gambar 4.3 Level 1 Proses Rawat Jalan



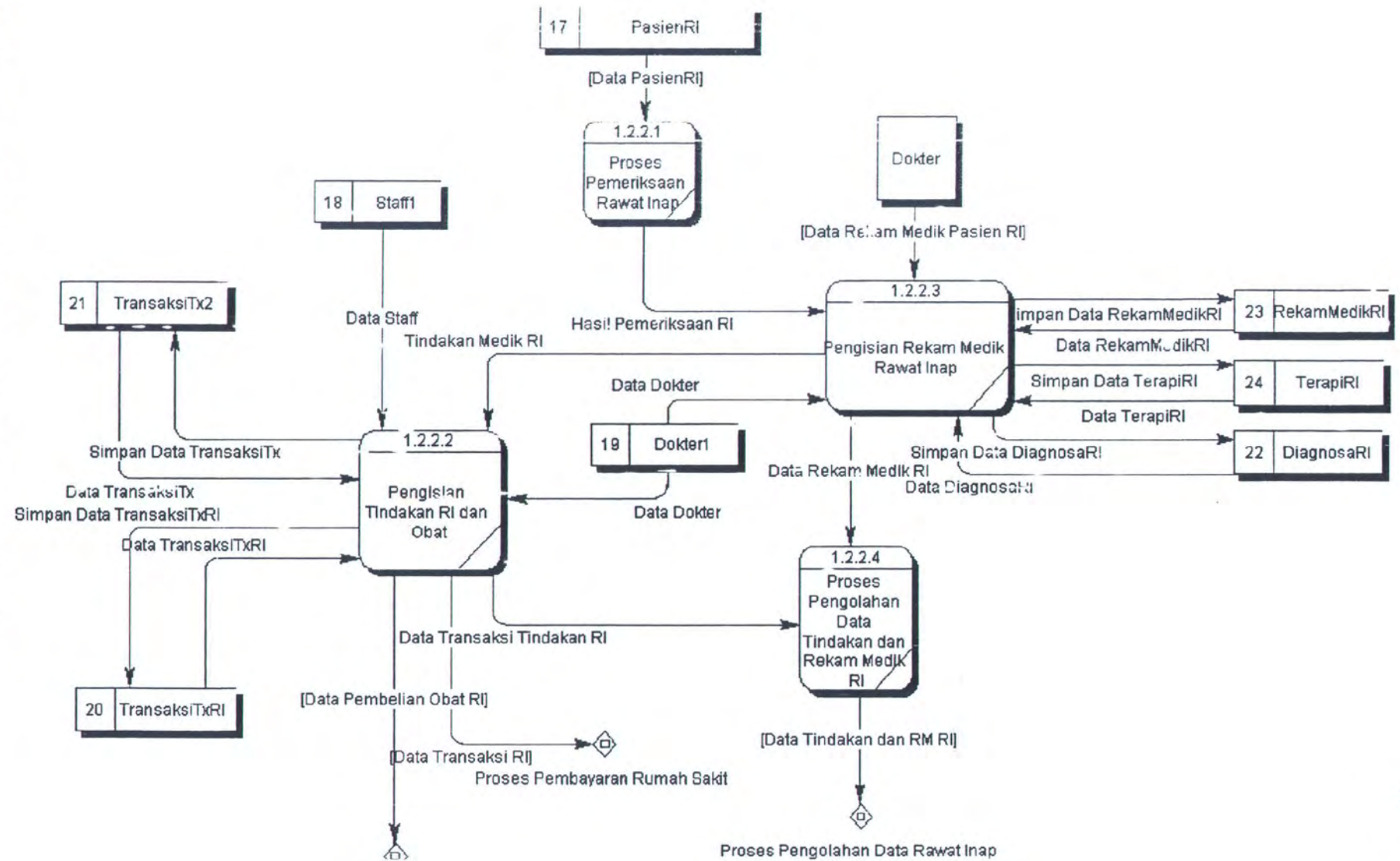
Gambar 4.4 Level 2 Proses Tindakan dan Rekam Medik Rawat Jalan



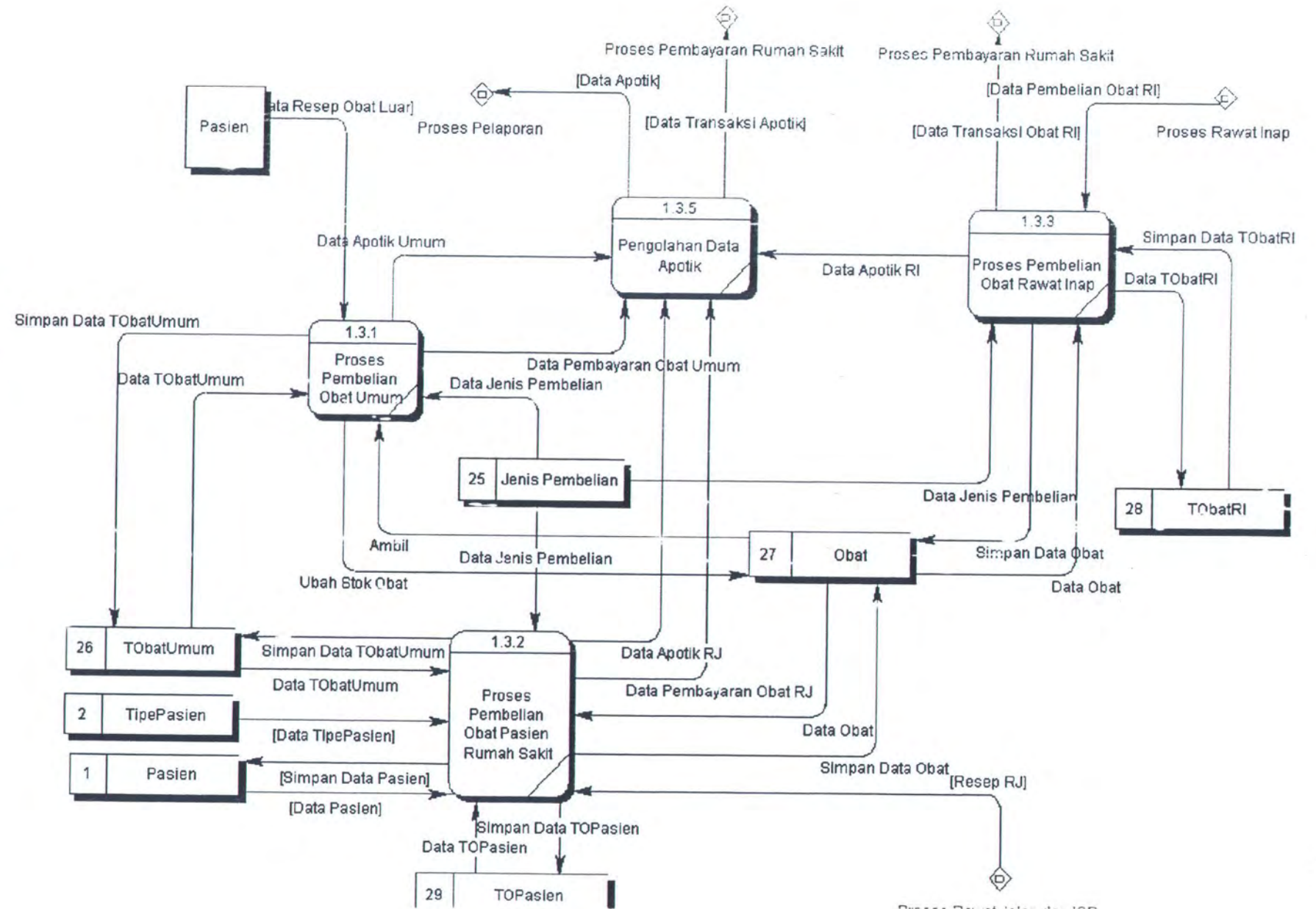
Gambar 4.5 Level 1 Proses Rawat Inap



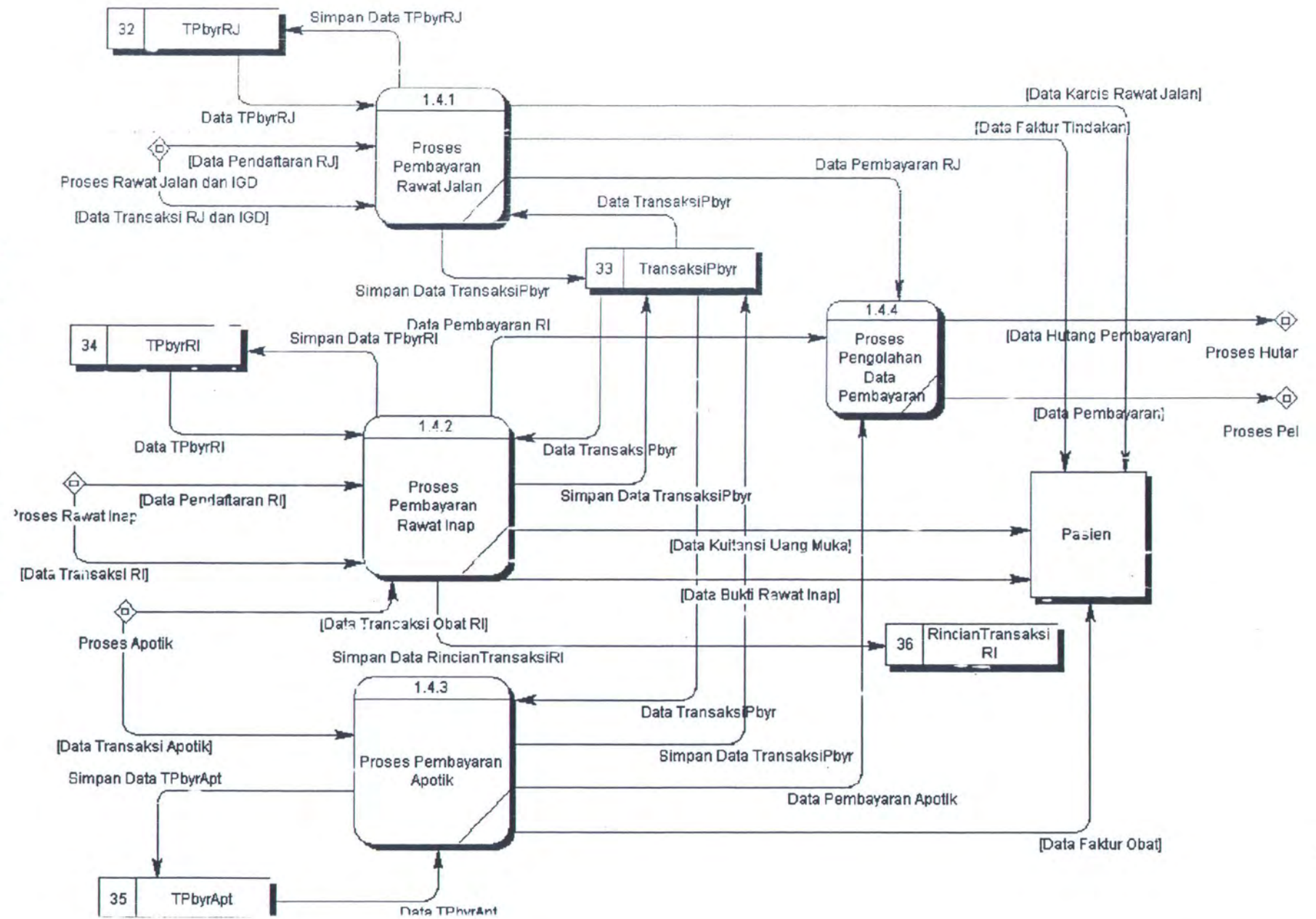
Gambar 4.6 Level 2 Proses Tindakan dan Rekam Medik Rawat Inap



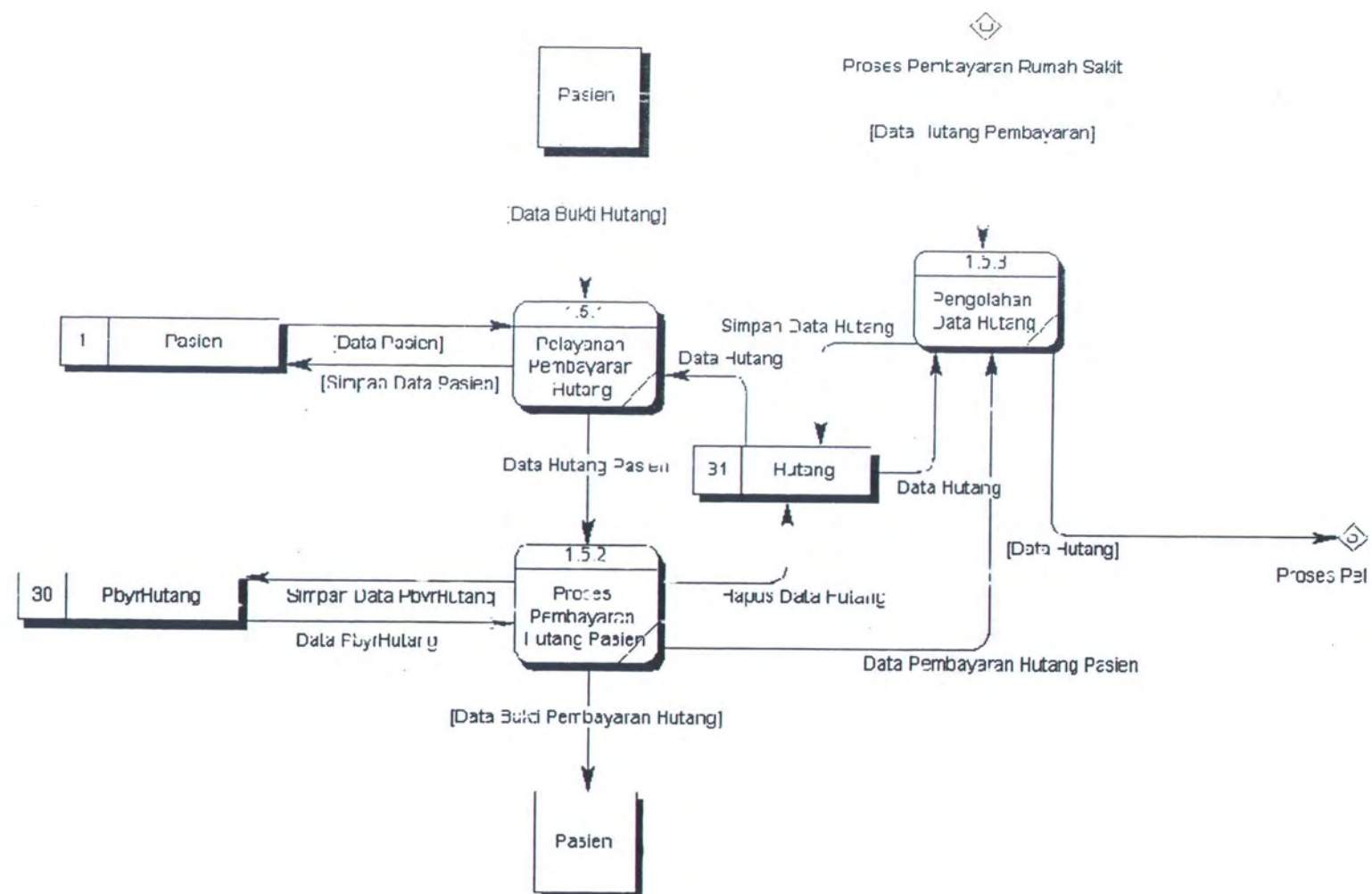
Gambar 4.7 Level 1 Proses Apotik



Gambar 4.8 Level 1 Proses Pembayaran



Gambar 4.9 Level 1 Proses Hutang Pasien



4.1.2 Pembuatan Conceptual Data Model

Dari proses-proses bisnis serta DFD diatas dikembangkan lagi dengan ke dalam bentuk Conceptual Data Model yang diagramnya dapat pada di halaman berikutnya.

4.2 Desain Sistem

Dalam tahapan ini akan dibahas pengembangan desain dan implementasi sistem dari tahap analisa diatas, yang dibagi dalam 3 tahapan, antara lain :

1. Files dan Databases

Dalam tahapan ini Conceptual Data Model yang dihasilkan dalam tahap analisa diatas, digenerate untuk menghasilkan tabel-tabel diperlukan dalam sistem.

2. Desain Forms

Dalam tahapan ini akan dihasilkan desain-desain Form serta kegunaannya dalam sistem.

3. Desain Reports

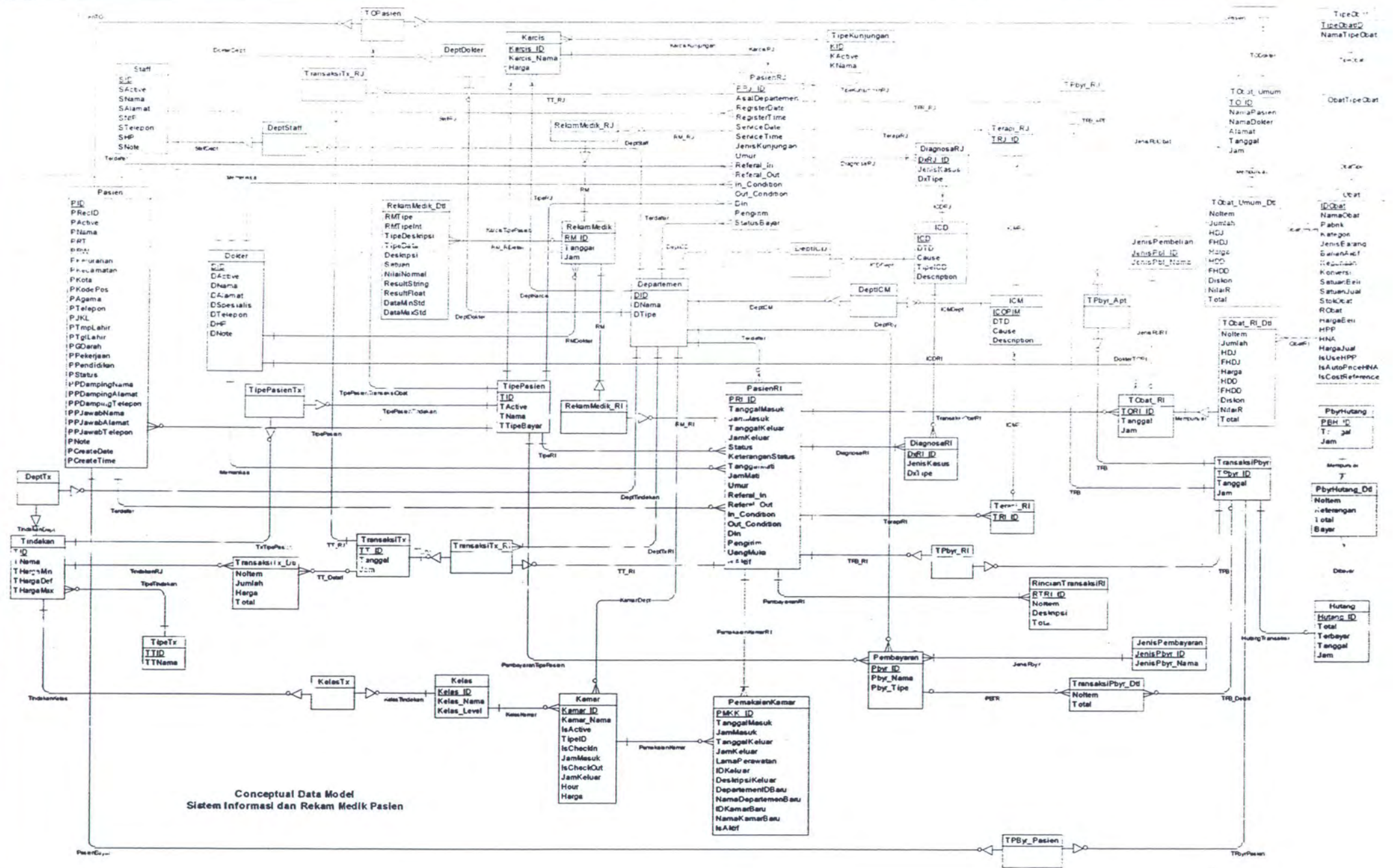
Dalam tahapan ini akan dihasilkan desain-desain Reports serta kegunaannya dalam sistem.

4. Desain Menu Sistem

Dalam tahapan ini akan dihasilkan rancangan menu dari sistem.

5. Spesifikasi tiap proses

Dalam tahapan ini akan diterangkan algoritma setiap proses yang dihasilkan dengan dipadukan dengan hasil desain diatas.



4.2.1 Files dan Database

Berikut ini adalah daftar tabel-tabel yang digunakan dalam sistem, hasil dari generate Conceptual Data Model :

Tabel 4.1 Daftar Table dalam Database

No.	Tabel	Fungsi
1	AktifitasJurnal	Mencatat segala aktifitas user
2	AutoNumbering	Membuat Unique ID
3	Departemen	Daftar Poli dan Ruang RS
4	DeptDokter	Departemen Tempat Dokter berada
5	DeptICD	Departemen Tempat ICD Ditampilkan
6	DeptICM	Departemen Tempat ICM Ditampilkan
7	DeptStaff	Departemen Tempat Staff Ditampilkan
8	DeptTemplate	Departemen Tempat Template Ditampilkan
9	DeptTindakan	Departemen Tempat Tindakan Ditampilkan
10	Deskripsi	Daftar Deskripsi untuk pendaftaran pasien
11	DiagnosaRI	Diagnosa Pasien Rawat Inap
12	DiagnosaRJ	Diagnosa Pasien Rawat Jalan
13	Dokter	Daftar Dokter
14	Hutang	Daftar Hutang
15	ICD	Daftar ICD
16	ICM	Daftar ICM
17	JenisPembayaran	Daftar Jenis Pembayaran Pasien
18	JenisPembelian	Daftar Jenis Pembelian Pasien
19	Kamar	Daftar Kamar Rumah Sakit
20	Karcis	Daftar Karcis Rawat Jalan
21	Kelas	Daftar Kelas Kamar Rumah Sakit
22	KelasTindakan	Kelas Tempat Tindakan Ditampilkan
23	Obat	Daftar Obat
24	ObatTipeObat	Obat Mempunyai Tipe Obat
25	Pasien	Daftar Pasien
26	PasienRI	Daftar Pasien Rawat Inap
27	PasienRJ	Daftar Pasien Rawat Jalan
28	PbyrHutang_Dtl	Daftar Pembayaran Hutang Detail
29	PbyrHutang	Daftar Pernbayaran Hutang
30	PemakaianKamar	Daftar Pemakaian Kamar Pasien
31	Pembayaran	Daftar Pembayaran
32	RekamMedik_Dtl	Daftar Rekam Medik Detail
33	RekamMedik	Daftar Rekam Medik
34	RekamMedikRI	Daftar Rekam Medik Rawat Inap
35	RekamMedikRJ	Daftar Rekam Medik Rawat Jalan
36	RincianTransaksiRI	Daftar Rincian Transaksi Pasien Rawat Inap
37	RMResultString	Hasil Rekam Medik Pasien
38	RMTemplate	Rekam Medik Layout untuk dipilih

39	SetHargaJual	Pengesetan Harga Jual Otomatis
40	Staff	Daftar Staff
41	SystemVariable	Tabel untuk menyimpan setup form
42	Template	Layout Untuk Rekam Med'k
43	TerapiRI	Daftar Terapi Pasien Rawat Inap
44	TerapiRJ	Daftar Terapi Pasien Rawat Jalan
45	Tindakan	Daftar Tindakan
46	TipeKunjungan	Daftar Tipe Kunjungan Pasien
47	TipeObat	Daftar Tipe Obat
48	TipePasien	Daftar Tipe Pasien Rumah Sakit
49	TipePasienTindakan	Tindakan Sesuai dengan Tipe Pasien
50	TipeTindakan	Daftar Tipe Tindakan
51	TObatRI_Dtl	Daftar Pembelian Obat Rawat Inap Detail
52	TObatRI	Daftar Pembelian Obat Rawat Inap
53	TObatUmum_Dtl	Daftar Pembelian Obat Rawat Jalan Detail
54	TObatUmum	Daftar Pembelian Obat Rawat Jalan
55	TODokter	Daftar Resep Dokter
56	TOPasien	Daftar Transaksi Obat Pasien
57	TPbyrApt	Daftar Transaksi Pembayaran Apotek
58	TPbyrRJ	Daftar Transaksi Pembayaran Rawat Jalan
59	TPbyrRI	Daftar Transaksi Pembayaran Rawat Inap
60	TransaksiPbyr_Dtl	Daftar Transaksi Pembayaran Detail
61	TransaksiPbyr	Daftar Transaksi Pembayaran
62	TransaksiTx_Dtl	Daftar Transaksi Tindakan Detail
63	TransaksiTx	Daftar Transaksi Tindakan
64	TransaksiTxPI	Daftar Transaksi Tindakan Rawat Inap
65	TransaksiTxRJ	Daftar Transaksi Tindakan Rawat Jalan
66	TxDokter	Daftar Tindakan oleh Dokter
67	TxStaff	Daftar Tindakan oleh Staff
68	UserAccess	Daftar User Program
69	UserModule	Daftar Module Untuk User
70	UserPrivilege	Daftar Menu Untuk User

Adapun penjelasan dari sebagian tabel-tabel diatas dalam bentuk DDL

(Data Definition Language) adalah sebagai mara berikut :

1. Obat

Tabel ini berisi data-data Obat yang diperjual belikan di rumah sakit.

```
CREATE TABLE Obat
(
  IDObat          VARCHAR(32) PRIMARY KEY,
  NamaObat        VARCHAR(250),
  Pabrik          VARCHAR(250),
  Kategori        VARCHAR(32),
  JenisBarang     INT,
  BahanAktif      VARCHAR(250),
  Kegunaan        VARCHAR(250),
  Konversi        FLOAT8 DEFAULT 0,
```

```

SatuanBeli          VARCHAR(32),
SatuanJual          VARCHAR(32),
RObat              FLOAT8 DEFAULT 0,
StokObat           FLOAT8 DEFAULT 0,
HargaBeli          FLOAT8 DEFAULT 0,
HPP               FLOAT8 DEFAULT 0,
HNA               FLOAT8 DEFAULT 0,
Margin            FLOAT8 DEFAULT 0,
IsUseHPP           INTEGER,
IsAutoPriceHNA     INTEGER,
IsCostReference    INTEGER,
HargaJual         FLOAT8 DEFAULT 0
);
GRANT ALL PRIVILEGES ON Obat TO PUBLIC;

```

2. Tindakan

Tabel ini berisi data-data Tindakan Pasien yang ada di rumah sakit

```

CREATE TABLE Tindakan
(
  TID          VARCHAR(32) PRIMARY KEY,
  TNama       VARCHAR(250),
  TType       VARCHAR(32) REFERENCES TipeTindakan
              ON UPDATE CASCADE ON DELETE SET NULL,
  THargaMin   FLOAT8 DEFAULT 0,
  THargaDef   FLOAT8 DEFAULT 0,
  THargaMax   FLOAT8 DEFAULT 0
);
GRANT ALL PRIVILEGES ON Tindakan TO PUBLIC;

```

3. TipePasien

Tabel ini berisi data-data Tipe Pasien yang ada di rumah sakit, seperti

Pasien Umum, Pasien Karyawan Rumah Sakit, Pasien Askes, dan lain-lain

```

CREATE TABLE TipePasien
(
  TID          VARCHAR(32) PRIMARY KEY,
  TActive      CHAR(1) DEFAULT 'Y',
  TNama       VARCHAR(250),
  TTypeBayar   VARCHAR(10)
);
GRANT ALL PRIVILEGES ON TipePasien TO PUBLIC;

```

Field TTypeBayar diatas menunjukkan Tipe Pembayaran pasien yang dibedakan menjadi 2 bagian 0: untuk Cash, 1: untuk Piutang.


4. Pasien

Tabel ini berisi data-data pasien yang terdaftar di rumah sakit

```

CREATE TABLE Pasien
(
  PRecID      INT,
  PActive     CHAR(1) DEFAULT 'Y',

```

```

PID          VARCHAR(32) PRIMARY KEY,
PNama        VARCHAR(250),
PAalamat     VARCHAR(250),
PRT          INT2 DEFAULT 0,
PRW          INT2 DEFAULT 0,
PKelurahan   VARCHAR(50),
PKecamatan   VARCHAR(50),
PKota        VARCHAR(50),
PKodePos     VARCHAR(20),
PAgama      VARCHAR(20),
PTelepon     VARCHAR(20),
PJKL         CHAR(1),
PTmpLahir    VARCHAR(50),
PTglLahir    DATE DEFAULT CURRENT_DATE,
PUmur        FLOAT4,
PGDarah      VARCHAR(2),
PPekerjaan   VARCHAR(50),
PPendidikan  VARCHAR(20),
PStatus      CHAR(1) DEFAULT 'N',
PPDampingNama VARCHAR(250),
PPDampingAlamat VARCHAR(250),
PPDampingTelepon VARCHAR(50),
PPJawabNama  VARCHAR(250),
PPJawabAlamat VARCHAR(250),
PPJawabTelepon VARCHAR(50),
PNote        VARCHAR(250),
PCreateDate  DATE DEFAULT CURRENT_DATE,
PCreateTime  TIME DEFAULT CURRENT_TIME,
PTipePasien  VARCHAR(32) REFERENCES TipePasien
ON UPDATE CASCADE ON DELETE SET NULL,
UNIQUE(PReclID)
);
CREATE INDEX MPasien_idx ON Pasien (PNama, PAalamat, PKelurahan, PKecamatan);
GRANT ALL PRIVILEGES ON Pasien TO PUELIC;

```

5. Departemen

Tabel ini berisi data-data poliklinik atau ruangan perawatan yang terdapat di rumah sakit

```

CREATE TABLE Departemen
(
  DID          VARCHAR(32) PRIMARY KEY,
  DNama        VARCHAR(250),
  DTipe        INT DEFAULT 0
);
GRANT ALL PRIVILEGES ON Departemen TO PUBLIC;

```

Field DTipe diatas menunjukkan Tipe Departemen itu berada yang dibedakan menjadi 3 bagian 0: untuk Rawat Jalan, 1: untuk Rawat Inap, 2: untuk Rawat Darurat.

6. TObatUmum

Tabel ini berisi data-data transaksi obat pasien rawat jalan dan transaksi obat oleh pasien luar

```

CREATE TABLE TransaksiObatUmum
(
  TO_ID          VARCHAR(32) NOT NULL PRIMARY KEY,
  NamaPasien     VARCHAR(250),
  NamaDokter     VARCHAR(250),
  Alamat         VARCHAR(250),
  JenisPbl_ID    VARCHAR(32) REFERENCES JenisPembelian
                ON UPDATE CASCADE ON DELETE SET NULL,
  Tanggal       DATE DEFAULT CURRENT_DATE,
  Jam           TIME DEFAULT CURRENT_TIME
);
GRANT ALL PRIVILEGES ON TransaksiObatUmum TO PUBLIC;

```

7. PasienRI

Tabel ini berisi daftar pasien yang pernah mendaftar rawat inap

```

CREATE TABLE PasienRI
(
  PRI_ID          VARCHAR(32) PRIMARY KEY,
  PasienID        VARCHAR(32) REFERENCES Pasien
                ON UPDATE CASCADE ON DELETE SET NULL,
  DepartemenID    VARCHAR(32) REFERENCES Departemen
                ON UPDATE CASCADE ON DELETE SET NULL,
  PasienTipeID    VARCHAR(32) REFERENCES TipePasien
                ON UPDATE CASCADE ON DELETE SET NULL,
  DokterID        VARCHAR(32) REFERENCES Dokter
                ON UPDATE CASCADE ON DELETE SET NULL,
  Kamar_ID        VARCHAR(32),
  TanggalMasuk    DATE,
  JamMasuk        TIME,
  TanggalKeluar    DATE,
  JamKeluar       TIME,
  Status          INT DEFAULT 0,
  KeteranganStatus VARCHAR(250),
  TanggalMati     DATE,
  JamMati         TIME,
  Umur            FLOAT4,
  Referral_In     VARCHAR(250),
  Referral_Out    VARCHAR(250),
  Pengirim        VARCHAR(250),
  DIn             VARCHAR(250),
  In_Condition    VARCHAR(250),
  Out_Condition   VARCHAR(250),
  UangMuka        FLOAT8,
  IsAktif         CHAR(1) DEFAULT 'Y'
);
GRANT ALL PRIVILEGES ON PasienRI TO PUBLIC;

```

8. TransaksiTx

Tabel ini berisi data-data transaksi tindakan baik rawat jalan maupun rawat inap.

```

CREATE TABLE TransaksiTx
(
  TT_ID          VARCHAR(32) NOT NULL PRIMARY KEY,
  TPasien_ID     VARCHAR(32) REFERENCES TipePasien
                ON UPDATE CASCADE ON DELETE SET NULL,
  Tanggal       DATE DEFAULT CURRENT_DATE,
  Jam           TIME DEFAULT CURRENT_TIME
);
GRANT ALL PRIVILEGES ON TransaksiTx TO PUBLIC;

```


9. RekamMedik

Tabel ini berisi data-data rekam medik pasien.

```
CREATE TABLE RekamMedik
(
  RM_ID          VARCHAR(32) NOT NULL PRIMARY KEY,
  Dokter_ID      VARCHAR(32) REFERENCES Dokter
                  ON UPDATE CASCADE ON DELETE SET NULL,
  Tanggal        DATE DEFAULT CURRENT_DATE,
  Jam            TIME DEFAULT CURRENT_TIME
);

GRANT ALL PRIVILEGES ON RekamMedik TO PUBLIC;
```

10. TransaksiPbyr

Tabel ini berisi data-data transaksi pembayaran yang dilakukan oleh pasien.

```
CREATE TABLE TransaksiPbyr
(
  TPbyr_ID       VARCHAR(32) NOT NULL PRIMARY KEY,
  Tanggal        DATE DEFAULT CURRENT_DATE,
  Jam            TIME DEFAULT CURRENT_TIME
);

GRANT ALL PRIVILEGES ON TransaksiPbyr TO PUBLIC;
```

Untuk penjelasan lebih lanjut dari daftar tabel di atas dapat dilihat pada lampiran pada bagian belakang buku ini.

4.2.2 Desain Forms

Untuk melakukan aktifitas-aktifitas dalam sistem dibutuhkan form-form input output sebagai interfaces dan program coding untuk menghubungkan database dengan sistem yang akan dibangun. Adapun form-form yang digunakan adalah sebagai berikut :

Tabel 4.2 Daftar Form

No.	Form	Fungsi
1	SUB_AKTIFITAS_JURNAL	Menampilkan aktifitas-aktifitas user
2	SUB_AKTIFITAS_JURNAL_SETUP	Setup dari Aktifitas Jurnal
3	SUB_CETAK_KARCIS	Form untuk mencetak karcis rawat

		jalan
4	SUB_DAFTAR_HUTANG	Menampilkan Daftar Hutang
5	SUB_DAFTAR_PASIEN_RAWAT_INAP	Menampilkan Daftar Pasien Rawat Inap
6	SUB_DAFTAR_PASIEN_RAWAT_JALAN	Menampilkan Daftar Pasien Rawat jalan
7	SUB_DAFTAR_PASIEN_SETUP	Setup dari Daftar Pasien Rawat Jalan
8	SUB_DAFTAR_PASIEN_SETUP_INAP	Setup dari Daftar Pasien Rawat Inap
9	SUB_DEPT_DOKTER	Menampilkan Dokter Bekerja di Departemen
10	SUB_DEPT_STAFF	Menampilkan Dokter Bekerja di Departemen
11	SUB_DESIASE_DIALOG	Input Dialog ICD dan ICOPIM
12	SUB_DESIASE_MASTER	Menampilkan Daftar ICD dan ICOPIM
13	SUB_EXPORT_IMPORT_OBJECT	Untuk mengimport data dari DBF
14	SUB_HISTORY_PASIEN_RAWAT_INAP	Menampilkan Data Pasien Rawat Inap yang sudah keluar
15	SUB_HOSPITAL_ADMINISTRATION_STATISTIC	Menampilkan Data Statistik Pemakaian Kamar Rumah Sakit
16	SUB_INPUT_DATA_PASIEN	Input Dialog Data Pasien
17	SUB_INPUT_DIAGNOSA	Input Dialog Data Diagnosa
18	SUB_INPUT_PASIEN_KONSUL	Input Dialog Pasien Konsul
19	SUB_INPUT_PASIEN_RAWAT_INAP	Input Dialog Data Pasien Rawat Inap
20	SUB_INPUT_PASIEN_RAWAT_JALAN	Input Dialog Data Pasien Rawat Jalan
21	SUB_INPUT_TERAPI	Input Dialog Terapi Pasien
22	SUB_INPUT_TINDAKAN_STAFF	Input Dialog Tindakan Staff
23	SUB_KIUP	Menampilkan Data Pasien Rumah Sakit
24	SUB_LEASING_MANAGEMENT	Menampilkan Data Kamar Perawatan
25	SUB_LEASING_MANAGEMENT_DIALOG	Input Dialog Data Kamar
26	SUB_LIHAT_TRANSAKSI	Menampilkan Data-Data Transaksi Apotek
27	SUB_LIHAT_TRANSAKSI_SETUP	Setup dari Lihat Transaksi
28	SUB_MAIN	Form Utama program
29	SUB_MASTER_DEPARTEMEN	Menampilkan Data Departemen
30	SUB_MASTER_DEPARTEMEN_DIALOG	Input Dialog Data Departemen
31	SUB_MASTER_DESKRIPSI	Menampilkan Data Deskripsi
32	SUB_MASTER_DESKRIPSI_DIALOG	Input Dialog Data Deskripsi
33	SUB_MASTER_DOKTER	Menampilkan Data Dokter
34	SUB_MASTER_JENIS PEMBAYARAN	Menampilkan Data Jenis Pembayaran
35	SUB_MASTER_JENIS PEMBAYARAN_DIALOG	Input Dialog Data Jenis Pembayaran
36	SUB_MASTER_JENIS PEMBELIAN	Menampilkan Data Jenis Pembelian
37	SUB_MASTER_JENIS PEMBELIAN_DIALOG	Input Dialog Data Jenis Pembelian
38	SUB_MASTER_KARCIS	Menampilkan Data Karcis Rawat Jalan
39	SUB_MASTER_KARCIS_DIALOG	Input Dialog Data Karcis
40	SUB_MASTER_KELAS	Menampilkan Data Kelas Perawatan
41	SUB_MASTER_KELAS_DIALOG	Input Dialog Data Kelas
42	SUB_MASTER_OBAT	Menampilkan Data Obat
43	SUB_MASTER PEMBAYARAN	Menampilkan Data Pembayaran

		Pasien
44	SUB_MASTER PEMBAYARAN_DIALOG	Input Dialog Data Pembayaran
45	SUB_MASTER_RMTEMPLATE_DIALOG	Input Dialog Template Rekam Medik
46	SUB_MASTER_STAFF	Menampilkan Data Staff Rumah Sakit
47	SUB_MASTER_TEMPLATE	Menampilkan Data Template Rekam Medik
48	SUB_MASTER_TINDAKAN	Menampilkan Data Tindakan Rumah Sakit
49	SUB_MASTER_TIPE_KUNJUNGAN	Menampilkan Data Tipe Kunjungan Pasien
50	SUB_MASTER_TIPE_KUNJUNGAN_DIALOG	Input Dialog Data Tipe Kunjungan
51	SUB_MASTER_TIPE_OBAT	Menampilkan Data Tipe Obat
52	SUB_MASTER_TIPE_OBAT_DIALOG	Input Dialog Data Tipe Obat
53	SUB_MASTER_TIPE_PASIEN	Menampilkan Data Tipe Pasien
54	SUB_MASTER_TIPE_PASIEN_DIALOG	Input Dialog Data Tipe Pasien
55	SUB_MASTER_TIPE_TINDAKAN	Menampilkan Data Tipe Tindakan
56	SUB_MASTER_TIPE_TINDAKAN_DIALOG	Input Dialog Data Tipe Tindakan
57	SUB_MEDICAL_RECORD_FREEFORM	Input Rekam Medik Free Form
58	SUB_PASIEN_REREGISTRASI	Form Mernilih Data Pasien
59	SUB_PEMAKAIAN_KAMAR	Menampilkan Data Pemakaian Kamar Rawat Inap
60	SUB PEMBAYARAN HUTANG	Form Transaksi Pembayaran Hutang
61	SUB PEMBAYARAN HUTANG SETUP	Setup Form Transaksi Pembayaran Hutang
62	SUB PEMBAYARAN HUTANG VIEW	Menampilkan Data Pembayaran Hutang
63	SUB PEMBAYARAN RAWAT INAP VIEW	Menampilkan Data Transaksi Pembayaran Rawat Inap
64	SUB PEMBAYARAN TRANSAKSI	Form Transaksi Pembayaran Rawat Inap
65	SUB PEMBAYARAN TRANSAKSI SETUP	Setup Form Transaksi Pembayaran Rawat Inap
66	SUB PENDAFTARAN PASIEN RAWAT INAP	Form Pendaftaran Pasien Rawat Inap
67	SUB PENDAFTARAN PASIEN RAWAT JALAN	Form Pendaftaran Pasien Rawat Jalan
68	SUB PENDAFTARAN PASIEN RI SETUP	Setup Form Pendaftaran Pasien Rawat Inap
69	SUB PENDAFTARAN PASIEN SETUP	Setup Form Pendaftaran Pasien Rawat Jalan
70	SUB PINDAH RUANGAN	Form Untuk Pindah Kamar dan Ruangan Rawat Inap
71	SUB_REPORT_DIAGNOSIS_DOKTER	Form Laporan Diagnosa Dokter
72	SUB_REKAM_MEDIK_PASIEN	Form Pengisian Rekam Medik Pasien
73	SUB_REKAM_MEDIK_PASIEN_VIEW	Menampilkan Data Rekam Medik Pasien
74	SUB_REPORT_APOTEK	Form Laporan Apotek
75	SUB_REPORT_KUNJUNGAN_RJ	Form Laporan Kunjungan Rawat Jalan
76	SUB_REPORT_MORBIDITAS	Form Laporan Morbiditas
77	SUB_REPORT_RANGKING_PENYAKIT	Form Laporan Ranging Penyakit
78	SUB_REPORT_REKAP_KARCIS	Form Laporan Rekap Karcis
79	SUB_REPORT_VISITING	Form Laporan Summary Kunjungan

		Rawat Jalan
80	SUB_RMRESULT_STRING_DIALOG	Dialog untuk Pengisian Template Rekam Medik
81	SUB_SELECT_OBAT	Form untuk memilih data obat
82	SUB_SELECT_PASIEN_REGISTRASI	Form untuk memilih data pasien rawat jalan
83	SUB_SELECT_PASIEN_REGISTRASI_INAP	Form untuk memilih data pasien rawat inap
84	SUB_SELECT_STAFF_DOKTER	Form untuk memilih data staff dan dokter
85	SUB_SELECT_TEMPLATE	Form untuk memilih data template rekam medik
86	SUB_SELECT_TINDAKAN	Form untuk memilih data tindakan
87	SUB_SET_HARGA_TIPE_PASIEN_DIALOG	Form untuk mengatur harga obat apotek
88	SUB_SYSTEM_MAINTAIN	Form untuk perawatan database
89	SUB_SYSTEM_SETUP	Setup form utama
90	SUB_TINDAKAN_PASIEN	Form Transaksi Tindakan Pasien
91	SUB_TINDAKAN_PASIEN_VIEW	Menampilkan Data Tindakan Pasien
92	SUB_TRANSAKSI_OBAT	Form Transaksi Obat Rawat Jalan
93	SUB_TRANSAKSI_OBAT_INAP	Form Transaksi Obat Rawat Inap
94	SUB_TRANSAKSI_OBAT_INAP_REVIEW	Menampilkan Transaksi Obat Rawat Inap lama
95	SUB_TRANSAKSI_OBAT_SETUP	Setup Form Transaksi Obat Rawat Jalan
96	SUB_TRANSAKSI_OBAT_SETUP_INAP	Setup Form Transaksi Obat Rawat Inap
97	SUB_USER_AUTHORITY	Menampilkan data user yang melakukan aktifitas
98	SUB_USER_LOGIN_DIALOG	Form Login untuk masuk ke program
99	SUB_USER_MANAGER	Form Untuk pengaturan user program
100	SUB_USER_MANAGER_DIALOG	Input Dialog user manager
101	SUB_USER_PASSWORD	Form untuk mengganti password user
102	SUB_USER_PRIVILEGE_DIALOG	Input Dialog untuk membuat menu baru
103	SUB_USER_PRIVILEGE_EDITOR	Input Dialog untuk mengubah menu
104	SUB_XINITIALIZE	Form inisialisasi awal program

Berikut ini adalah desain beberapa form diatas, adapun desainnya adalah sebagai mana berikut :

1. Form Login

Form ini digunakan user untuk masuk ke dalam sistem dengan memasukkan IP Server, User ID dan Password.

Gambar 4.11 Form User Login

2. Form User Manager

Form ini digunakan untuk membuat user di dalam aplikasi sistem informasi ini.

Gambar 4.12 Form User Manager

User manager juga digunakan untuk membuat daftar menu dalam form utama yang diinginkan yang otomatis tersambung dengan modul-modul didalam program. Hanya ketiga user awal saja yang bisa membuat tambahan menu untuk aplikasi.

3. Form Input Pasien Baru

Form ini digunakan untuk memasukkan data-data dari pasien baru yang belum pernah mendaftar di rumah sakit.

Input Data Pasien

RM

Nama

JkL

Status

Tmp. Lhr

Tgl. Lhr

Umur

Gol Darah

Alamat

RT / RW

Desa/Kecurahan

Kecamatan

Kode Pos

Kota

Telepon

Agama

Pendidikan

Pekerjaan

Catatan

Suami

Telepon

Alamat

Kontak

Telepon

Alamat

Batal

Simpan

Gambar 4.13 Form Input Pasien Baru

4. Form Input Pasien Rawat Jalan

Form ini digunakan untuk memasukkan data pasien yang akan mendaftar rawat jalan atau instalasi gawat darurat. Disini dimasukkan data utama pasien dan pilihan poliklinik, tipe pasien, dan karcis sesuai dengan permintaan pasien.

Data Pasien Rawat Jalan

RM

Nama

Lahir

Alamat

RT / RW

Desa / Kecamatan

Kota

Gol Darah

Poli

Kategori

T Bayar

Karcis

Biaya

Cara Medik.

Oleh

Dik

Kondisi Med..

Batal

Input

Simpan Data Pasien

Gambar 4.14 Form Input Pasien Rawat Jalan

5. Form Input Pasien Rawat Inap

Form ini digunakan untuk memasukkan data-data pasien rawat inap. Hampir sama dengan Input Pasien Rawat Jalan, disini juga

dimasukkan data utama pasien dan pilihan ruangan dan kamar yang ditempati, tipe pasien, uang muka perawatan, dan dokter yang merawat.

Gambar 4.15 Form Input Pasien Rawat Inap

Ketika data disimpan, form akan secara otomatis mencatat data pemakaian kamar di tabel Pemakaian Kamar. Dimana Tabel Pemakaian Kamar berguna untuk perhitungan statistik pemakaian kamar rawat inap.

6. Form Pindah Kamar

Form dialog ini digunakan untuk memindah atau mengeluarkan pasien rawat inap dari kamar atau ruangan perawatan.

Gambar 4.16 Form Input Pasien Pindah Kamar

7. Form Dialog Rekam Medik

Form ini digunakan untuk mengisi rekam medik pasien. Untuk mengisi Form ini digunakan dua metode yaitu dengan menggunakan Free Form yaitu mengisi bebas atau menggunakan Layout Pemeriksaan yang sudah ada.

Untuk rawat jalan pada form ini juga digunakan untuk mengisi kondisi pasien dan keadaan pasien ketika masuk dan keluar rumah sakit.

Gambar 4.17 Form Input Rekam Medik Pasien

8. Form Tindakan Pasien

Form ini digunakan untuk memasukkan transaksi tindakan yang dilakukan oleh dokter terhadap pasien rumah sakit. Terdapat dua bagian pada form ini yaitu :

1. Untuk pengisian Tindakan Medik Pasien

Bagian ini digunakan untuk memasukkan daftar transaksi tindakan serta dokter atau staff yang melakukan tindakan tersebut.

TINDAKAN MEDIK PASIEN

Nama Pasien Tipe Pasien

Tanggal / / 3:02:52 PM Departemen

Tindakan Medik Pasien | Pembayaran

No	ID	Nama	Jml	Harga

No Item 0

ID

Jumlah 0 Harga 0.00

Total 0.00

Pelaksana Tindakan

ID	Nama Staff	Tipe Staff	Tipe Tx

Tambah Hapus

Total Transaksi 0.00

Tambah Ubah Hapus Batal OK Simpan Tampil Cetak Desain Keluar

Gambar 4.18 Form Input Transaksi Tindakan Pasien

2. Untuk pengisian Pembayaran oleh pasien

TINDAKAN MEDIK PASIEN

Nama Pasien Tipe Pasien

Tanggal / / 3:02:52 PM Departemen

Tindakan Medik Pasien | Pembayaran

No	ID	Nama Pembayaran	Total

No Item 0

ID

Total 0.00

Total Transaksi 0.00

Tambah Ubah Hapus Batal OK Simpan Tampil Cetak Desain Keluar

Gambar 4.19 Form Input Pembayaran Transaksi Tindakan Pasien

Bagian Form ini bisa digunakan untuk mengisi pembayaran yang berbeda-beda. Jumlah tindakan yang harus dibayar bisa dibagi-pembayarannya, baik dalam bentuk tunai atau piutang.

9. Form Input Diagnosa Pasien

Form ini digunakan untuk memasukkan diagnosa dari pasien setelah diperiksa oleh dokter. Terdapat beberapa tipe dari diagnosa yang dimasukkan oleh dokter, antara lain :

1. Dx In : Diagnosa Masuk Pasien
2. Dx PreOp : Diagnosa Sebelum Operasi
3. Dx PostOp : Diagnosa Setelah Operasi
4. Dx PA : Diagnosa Patologi Anatomi
5. Dx Out : Diagnosa Keluar Pasien

Untuk pasien rawat jalan, biasanya hanya diisi diagnosa keluar pasien saja.

Gambar 4.20 Form Input ICD Pasien

10. Form Penjualan Obat Apotik

Form ini digunakan untuk mengisi transaksi pembelian obat pasien di apotik baik pasien rumah sakit ataupun pasien luar rumah sakit. Cara pengisian form ini sama dengan form transaksi tindakan pasien. Namun dalam form ini user harus mengisi jenis pembelian oleh pasien.

Ketika mengisi jenis pembelian pasien, maka dengan otomatis harga yang keluar akan berubah, dikalikan dengan faktor pengali untuk harga jual yang ditentukan.

Gambar 4.21 Form Input Transaksi Obat Pasien

Untuk Bagian dari Pembayaran sama dengan Transaksi Tindakan Pasien. Untuk Pasien Luar, Nama dan Alamat serta Dokter harus diisi secara manual.

11. Form Pembayaran Pulang Rawat Inap

Form ini digunakan untuk pembayaran pasien ketika pulang dari rawat inap. Ketika nama pasien dipilih, form ini secara otomatis akan mengambil data-data transaksi tindakan yang pernah dilakukan oleh pasien rawat inap yang dikelompokkan berdasarkan tipe tindakan, transaksi obat rawat inap yang pernah dilakukan serta pemakaian kamar ketika pasien di rawat di rumah sakit.

Pembayaran Rawat Inap			
File View			
Nama Pasien		Tipe Pasien	
Alamat			
Nama Dokter			
Tanggal Masuk	/ / 19	52 PM	Tanggal Pulang / / 19 3:02:52 PM
Kamar		Kelas	
Transaksi Pasien Rawat Inap Pembayaran			
No.	Deskripsi	Total	
Total Transaksi		0.00	
Uang Muka		0.00	
Sisa Pembayaran		0.00	
Tambah	Ubah	Hapus	Batal
Simpan	Tampil	Cetak	Desain
Setup	Keluar		

Gambar 4.22 Form Pembayaran Pulang Rawat Inap

12. Form Statistik Rumah Sakit

Form ini menampilkan statistik pemakaian kamar rumah sakit berdasarkan kelas yang ada di rumah sakit. Adapun informasi yang ditampilkan adalah :

1. Bed Occupancy Rate (BOR) yang menunjukkan persentase rata-rata tempat tidur yang dihuni oleh penderita dengan perhitungan :

$$BOR = \frac{N}{B} \times 100$$

Keterangan :

N = Jumlah Lama Pemakaian Kamar Per Kelas dalam jangka waktu pengamatan

B = Jumlah Kamar Per Kelas dikalikan jumlah waktu pengamatan

28	RekapKarcisRawatJalan.frf	Laporan Rekap Karcis Rawat Jalan
29	RekapTindakanRawatJalan.frf	Laporan Rekap Tindakan Rawat Jalan
30	SummaryTransaction.frf	Laporan Summary Transaksi Obat Per Transaksi
31	SummaryTransactionByCustomer.frf	Laporan Summary Transaksi Obat Per Pasien
32	SummaryTransactionByDoctor.frf	Laporan Summary Transaksi Obat Per Dokter
33	TransactionApotek.frf	Laporan Transaksi Obat Per Transaksi
34	TransactionByCustomer.frf	Laporan Transaksi Obat Per Pasien
35	TransactionByDoctor.frf	Laporan Transaksi Obat Per Dokter

Adapun sebagian desain-desain dari daftar laporan-laporan diatas adalah sebagaimana berikut :

1. DaftarPasienRJ

Laporan ini digunakan untuk menampilkan daftar pasien yang rawat jalan yang berkunjung pada periode tertentu.

Report title

[CompanyIName]

[CompanyAddress]

[CompanyCity]

Daftar Kunjungan Pasien Rawat Jalan Tanggal [/TanggalMasuk]

Master header

NO. RM	NAMA	ALAMAT	KELURAHAN	KECAMATAN	S U	POLIKLINIK	TIPE PASIEN
--------	------	--------	-----------	-----------	-----	------------	-------------

Master data

[VDPasier]	[VNamePasien]	[VAlamat]	[VKelurahan]	[VKecamatan]	[VSUm]	[VDeptID]	[VTipePasien]
------------	---------------	-----------	--------------	--------------	--------	-----------	---------------

Master footer

[Jumlah Record : [VJumlahRecord]]

Gambar 4.24 Desain Laporan DaftarPasienRJ

2. MorbiditasReportRI

Laporan ini digunakan untuk menampilkan laporan morbiditas rawat inap per tahun untuk diserahkan kepada Dinas Kesehatan.

Page header

DATA KEADAAN MORBIDITAS PASIEN RAWAT INAP RUMAH SAKIT

[TAHUN] [VYear]

FORMULIR RI2a

Halaman [PAGE#]

NAMA RUMAH SAKIT : [CompanyName]

No. Kode RS :

3525010

NO URUT	NO BTD	NO BAYAR TERPERINCI	GOLONGAN SEBAB SAKIT	PASIEN KELUAR (MIDUP & MATI) MENURUT GOLONGAN UMUR								MENURUT SEX		Jml Pasien Keluar	Jml Pasien Kal. Mati
				0 < 28 MR	28 Mr < 1 Th	1 - 4 Th	5 - 14 Th	15 - 24 Th	25 - 44 Th	45 - 64 Th	65 + Th	Lk	Pr		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16

Master data

[No]	[DID]	[WID]	[Uwe]	[Uw1]	[Uw2]	[Uw3]	[Uw4]	[Uw5]	[Uw6]	[Uw7]	[Uw8]	[Uw9]	[Uw10]	[Uw11]	[Uw12]	[Uw13]	[Uw14]	[Uw15]	[Uw16]
------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	-------	--------	--------	--------	--------	--------	--------	--------

Master footer

JUMLAH				[SumUw1]	[SumUw2]	[SumUw3]	[SumUw4]	[SumUw5]	[SumUw6]	[SumUw7]	[SumUw8]	[SumUw9]	[SumUw10]	[SumUw11]	[SumUw12]	[SumUw13]	[SumUw14]	[SumUw15]	[SumUw16]
--------	--	--	--	----------	----------	----------	----------	----------	----------	----------	----------	----------	-----------	-----------	-----------	-----------	-----------	-----------	-----------

Gambar 4.25 Desain Laporan MorbiditasReportRI

3. RekapKarcisRawatJalan

Laporan ini digunakan untuk menampilkan rekap karcis rawat jalan yang dicetak setiap selesai jam pelayaran rawat jalan.

Master data

[CompanyName]

[CompanyAddress]

REKAPITULASI LAPORAN PENDAPATAN KARCIS HARIAN PELAYANAN RAWAT JALAN

Dari Tanggal [VStartDate] Sampai Dengan Tanggal [VEndDate]

Detail header

Poliklinik : [VNameDepartemen]

JENIS PASIEN	JUMLAH PASIEN			CARA PEMBAYARAN			KETERANGAN	
	Blm.Byr.	Bayar	Total	Tunai	Piutang	Total	Blm.Byr.	Bayar
[VTypePasiend]	BlmByr	nBayar	alPasien	[VBayarTunai]	[VBayarPiutang]	[VTotalBayar]	[VekumBayar]	[VBayar]
Total	xBlmByr	nPxByr	SumTotalPx	[VTotalByrTunai]	TotalByrPiutang	SumTotalBayar	[VTotaBlmByr]	[VTotalByr]

Detail footer

Yang melaporkan

Nip.

Gambar 4.26 Desain Laporan RekapKarcisRawatJalan

4. TransactionByCostumer

Laporan ini digunakan untuk menampilkan penjualan obat di apotek dalam periode tertentu berdasarkan tanggal pembelian obat.

Page header	<p>Laporan Penjualan [DeptID] [CompanyName] [CompanyCity] TANGGAL : [StartDate] Jam [StartTime] S/D TANGGAL : [EndDate] Jam [EndTime]</p>				
Master data					
Detail data	Tanggal Penjualan : [TransactDate]				
	Pasien : [RelationName], [RelationID] [Ext aObject] Dokter : [DokterName], [DokterID]				
Subdetail header	Kode Produk	Nama Produk	Jml	Harga	Total
Subdetail data	[ProductCode]	[ProductName]	[Quantity]	[Price]	[Total]
Subdetail footer					[Total] [SumTotal]
Detail footer					
Master footer	Total Transaksi [mTotalMaster]				

Gambar 4.27 Desain Laporan TransactionByCustomer

5. PoliReportDiagnosisDokter

Laporan ini digunakan untuk menampilkan daftar diagnosis dokter.

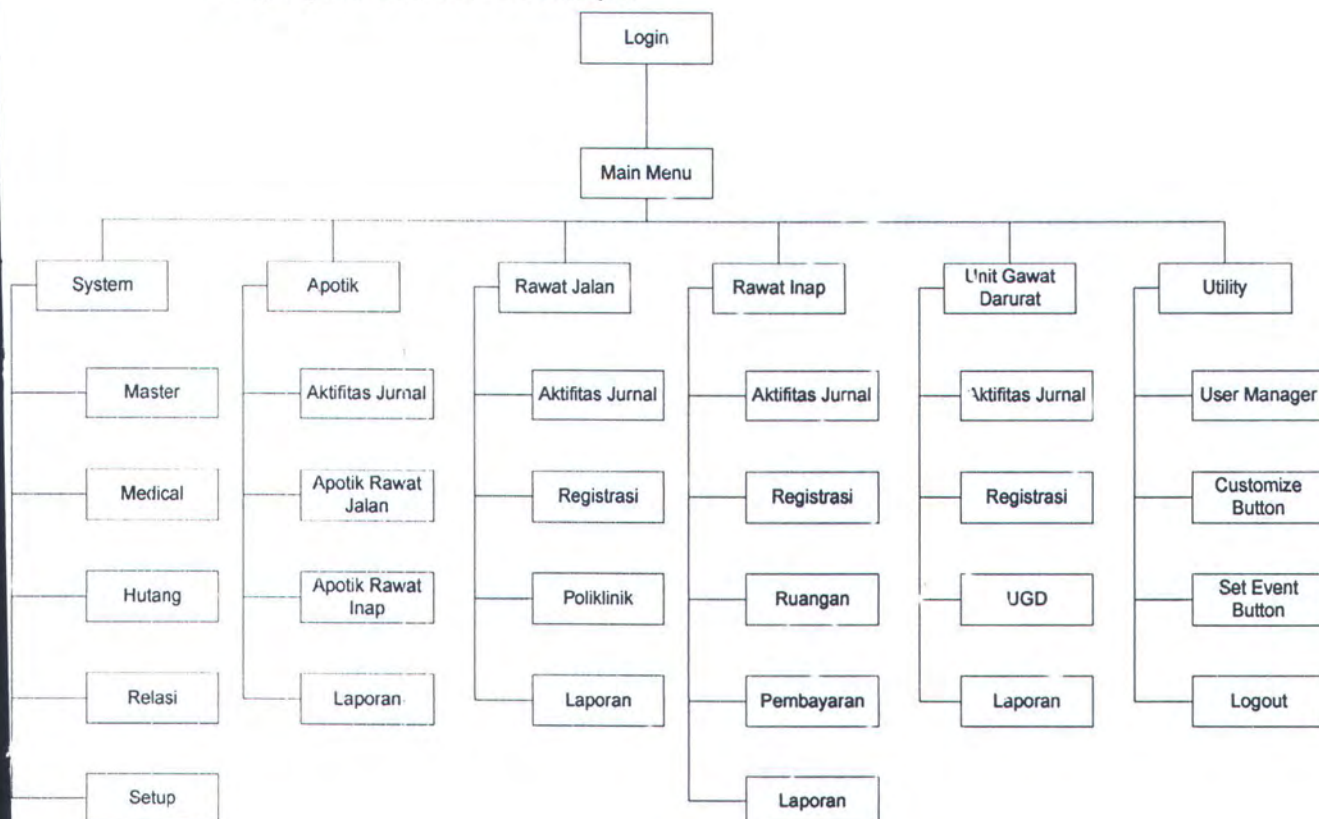
Page header	<p>LAPORAN DAFTAR PASIEN DOKTER [CompanyName] PERIODE TANGGAL : [VStartDate] S/D [VEndDate]</p>					
Master data	[Poliklinik : [VDepartmentName]]					
Detail data	[Dokter : [VMediatorName], [VMediatorID]]					
Subdetail header	Nomor	Pasien ID	Nama Pasien	Tgl. Dtg	ICD	Diagnose
Subdetail data	[Nomor]	[VOwnerID]	[VRelationName]	[VRegisterDate]	[VCD]	[VDiagnose]
Master header						

Gambar 4.28 Desain Laporan PoliReportDiagnosisDokter

Untuk desain laporan-laporan lain dapat dilihat pada lampiran di bagian belakang buku ini.

4.2.4 Desain Menu Sistem

Penulis mendesain menu yang disesuaikan dengan bagian-bagian dalam rumah sakit. Desain ini bisa dirubah lagi dengan karena menu dalam sistem ini bersifat dinamik Adapun desain menu utama dalam sistem ini terdapat pada halaman berikutnya.



Gambar 4.29 Desain Menu Root Aplikasi

4.2.5 Spesifikasi Proses

Proses-proses bisnis yang dihasilkan diatas mempunyai spesifikasi-spesifikasi dalam bentuk algoritma sebagai berikut :

1. Proses Rawat Jalan

- Proses Pendaftaran Rawat Jalan dan IGD :
 - Membuka Form Pendaftaran Rawat Jalan
 - Klik menu Baru untuk membuat pasien baru atau klik menu Re untuk pasien yang lama.
 - Untuk pasien baru setelah memilih menu Baru, akan muncul Form Input Data Pasien.
 - Ketika disimpan Form ini akan menyimpan inputan ke dalam Tabel Pasien.
 - Akan tampil Form Input Pasien Rawat Jalan.
 - Masukkan No. RM, Poliklinik, Kategori, Tipe Bayar, Tipe Pasien, Karcis, dan data-data yang lain.
 - Ketika disimpan, data inputan akan disimpan dalam Tabel PasienRJ.
- Proses Tindakan dan Rekam Medik Rawat Jalan dan IGD :
 - Membuka Form Daftar Pasien Rawat Jalan yang otomatis mengambil data dari Tabel PasienRJ.
 - Untuk memasukkan keterangan pasien, ubah data keterangan dan akan otomatis mengupdate Tabel PasienRJ.
 - Untuk mengubah data Diagnosa, klik pada Tambah Dx, Ubah Dx, dan Delete Dx yang otomatis mengupdate Tabel DiagnosaRJ.

- Untuk mengubah data Terapi, klik pada Tambah Terapi, Ubah Terapi, dan Delete Terapi yang otomatis mengupdate Tabel TerapiRJ.
- Untuk menambah data Rekam Medik Pasien, klik pada Rekam Medik yang akan memanggil Form Rekam Medik Pasien. Data yang dimasukkan dalam Form ini akan mengupdate Tabel RekamMedik, RekamMedik_Dtl, dan RekamMedikRJ.
- Untuk menambah data Transaksi Tindakan Pasien, klik pada Transaksi yang akan memanggil Form Tindakan Pasien. Pilih tindakan yang akan dimasukkan dari Tabel Tindakan. Data yang dimasukkan dalam Form ini akan mengupdate Tabel TransaksiTx, TransaksiTx_Dtl, dan TransaksiTxRJ.
- Proses Pengolahan Data Rawat Jalan :
 Dari tabel-tabel yang digunakan dalam proses rawat jalan akan dibuat beberapa view untuk memudahkan dalam pengambilan data untuk pembuatan laporan-laporan rawat jalan.

2. Proses Rawat Inap

- Proses Pendaftaran Rawat Inap :
 - Membuka Form Pendaftaran Rawat Inap.
 - Klik menu Baru untuk membuat pasien baru atau klik menu Re untuk pasien yang lama.
 - Untuk pasien baru setelah memilih menu Baru, akan muncul Form Input Data Pasien.

- Ketika disimpan Form ini akan menyimpan inputan ke dalam Tabel Pasien.
- Akan tampil Form Input Pasien Rawat Inap.
- Masukkan Tanggal dan Jam Pendaftaran, No. RM, DeptID, Uang Muka, TipePasien, Dokter, dan data-data yang lain.
- Masukkan data kamar dengan memilih dari Form Select Kamar dari Tabel Kamar yang akan disimpan secara otomatis dalam Tabel PemakaianKamar.
- Ketika disimpan, data inputan akan disimpan dalam Tabel PasienRI.
- Proses Tindakan dan Rekam Medik Rawat Inap :
 - Membuka Form Daftar Pasien Rawat Inap yang otomatis mengambil data dari Tabel PasienRI.
 - Untuk memasukkan keterangan pasien. ubah data keterangan dan akan otomatis mengupdate Tabel PasienRI.
 - Untuk mengubah data Diagnosa, klik pada Tambah Dx, Ubah Dx, dan Delete Dx yang otomatis mengupdate Tabel DiagnosaRI.
 - Untuk mengubah data Terapi, klik pada Tambah Terapi, Ubah Terapi, dan Delete Terapi yang otomatis mengupdate Tabel TerapiRI.
 - Untuk menambah data Rekam Medik Pasien, klik pada Rekam Medik yang akan memanggil Form Rekam Medik Pasien. Data

yang dimasukkan dalam Form ini akan mengupdate Tabel RekamMedik, RekamMedik_Dtl, dan RekamMedikRI.

- Untuk menambah data Transaksi Tindakan Pasien, klik pada Transaksi yang akan memanggil Form Tindakan Pasien. Pilih tindakan yang akan dimasukkan dari Tabel Tindakan Data yang dimasukkan dalam Form ini akan mengupdate Tabel TransaksiTx, TransaksiTx_Dtl, dan TransaksiTxRI serta akan mengupdate data TxDokter dan TxStaff.

- Proses Pengolahan Data Rawat Inap :

Dari tabel-tabel yang digunakan dalam proses rawat inap akan dibuat beberapa view untuk memudahkan dalam pengambilan data untuk pembuatan laporan-laporan rawat inap.

3. Proses Apotik

- Proses Pembelian Obat Umum :

- Membuka Form Transaksi Obat.
- Mengisi data pasien dan dokter secara manual.
- Memilih Jenis Pembelian dan Tipe Pasien
- Memilih obat-obat yang ditransaksikan dari Tabel Obat.
- Ketika data disimpan akan otomatis mengupdate Tabel TObatUmum, dan TObatUmum_Dtl.

- Proses Pembelian Obat Pasien Rumah Sakit :

- Membuka Form Transaksi Obat.

- Memilih data pasien dari Tabel Pasien dan data dokter dari Tabel Dokter.
- Memilih Jenis Pembelian dan Tipe Pasien
- Memilih obat-obat yang ditransaksikan dari Tabel Obat.
- Ketika data disimpan akan otomatis mengupdate Tabel TObatUmum, TObatUmum_Dtl, TOPasien dan TODokter.
- Proses Pembelian Obat Rawat Inap :
 - Membuka Form Transaksi Obat Rawat Inap.
 - Memilih data pasien rawat inap dari Tabel PasienRI dan data dokter dari Tabel Dokter.
 - Memilih Jenis Pembelian dan Tipe Pasien.
 - Memilih obat-obat yang ditransaksikan dari Tabel Obat.
 Ketika data disimpan akan otomatis mengupdate Tabel TobatRI dan TObatRI_Dtl.

• Proses Pengolahan Data Apotek :

Dari tabel-tabel yang digunakan dalam proses apotek akan dibuat beberapa view untuk memudahkan dalam pengambilan data untuk pembuatan laporan-laporan penjualan apotek.

4. Proses Pembayaran Rumah Sakit Pasien

- Proses Pembayaran Rawat Jalan dan IGD :
 - Untuk Karcis Rawat Jalan klik tombol Karcis pada Form Pendaftaran Rawat Jalan atau dengan membuka Form Cetak Karcis



yang akan mencetak Kwitansi Karcis Rawat Jalan. Aplikasi akan mengupdate data PasienRJ.

- Untuk membayar Tindakan Rawat Jalan pada Form Tindakan Pasien, setelah memasukkan data tindakan dalam proses tindakan pasien diatas klik tab dialog Pembayaran. Kemudian masukkan data pembayaran yang diambil dari Tabel Pembayaran. Ketika disimpan akan mengupdate Tabel TransaksiPbyr, TransaksiPbyr_Dtl, dan TPbyrRJ.
- Proses Pembayaran Rawat Inap :
 - Untuk membayar Uang Muka Rawat Inap klik tombol Kwitansi pada Form Pendaftaran Rawat Inap yang akan mengambil data dari Tabel PasienRI yang akan mencetak Kwitansi Uang Muka Rawat Inap. Aplikasi akan mengupdate data PasienRI.
 - Untuk membayar transaksi pasien selama pasien dirawat, buka Form Pembayaran Pulang Rawat Inap. Pilih pasien dari Tabel PasienRI, form akan otomatis mengambil data-data dari TransaksiTxRI, TransaksiObatRI dan PemakaianKamar. Untuk membayar klik pada Tab Dialog Pembayaran, kemudian pilih data pembayaran dari Tabel Pembayaran. Ketika disimpan aplikasi akan mengupdate Tabel RincianTransaksiRI, TransaksiPbyr, TransaksiPbyr_Dtl, dan TpbyrRI.
- Proses Pembayaran Obat :

Untuk Pembayaran Transaksi Obat Umum dan Pasien, setelah memasukkan data obat dalam proses transaksi obat umum dan pasien diatas klik tab dialog Pembayaran. Kemudian masukkan data pembayaran yang diambil dari Tabel Pembayaran. Ketika disimpan akan mengupdate Tabel TransaksiPbyr, TransaksiPbyr_Dtl, dan TPbyrApt.

Apabila ada pembayaran yang bertipe hutang, maka aplikasi akan otomatis mengupdate Tabel Hutang.

5. Proses Hutang Pasien

- Proses Pelayanan Pembayaran Hutang :
 - Membuka Form Pembayaran Hutang.
 - Memilih data pasien dari Tabel Pasien.
 - Memilih hutang-hutang yang akan dibayar dari tabel Hutang.
- Proses Pembayaran Hutang Pasien :

Ketika disimpan aplikasi akan mengupdate Tabel Hutang.
- Proses Pengolahan Data Hutang :

Dari Tabel Hutang akan dibuat beberapa view untuk memudahkan dalam pengambilan data untuk pembuatan laporan-laporan Hutang dan Pembayaran Hutang.

6. Proses Pelaporan

Dalam proses ini, form-form pelaporan akan mengambil data dari tabel-tabel diatas dan view-view dari proses pengolahan-pengolah data diatas untuk menghasilkan laporan-laporan rumah sakit.

BAB V

IMPLEMENTASI DAN UJI COBA SISTEM

Pada Bab 5 ini akan dibahas implementasi dan uji coba aplikasi Sistem Informasi Pasien dan Rekam Medik.

5.1 Implementasi Sistem

Tujuan implementasi adalah untuk menghasilkan aplikasi Sistem Informasi Pasien dan Rekam Medik dengan menambahkan proses coding untuk menghubungkan hasil dari tahap-tahapan sebelumnya.

Class-class yang terdapat dalam proses implementasi ini, dihasilkan dari seluruh internal entity dalam Proses Bisnis pada Bab IV diatas, yang dikelompokkan lagi ke dalam dbmodule-dbmodule buatan Penulis. Adapun dbmodule-dbmodule yang digunakan tidak berhubungan langsung dengan proses bisnis, hanya untuk memudahkan Penulis dalam pembuatan aplikasi Sistem Informasi Pasien dan Rekam Medik.

Proses-proses coding utama dalam program terletak dalam dbmodule-dbmodule yang berisi Class-Class yang dapat dipanggil secara langsung oleh Form-form diatas. Adapun daftar dbmodule dalam aplikasi ini adalah :

Tabel 5.1 Daftar DBModule

No.	Nama DbModule	Deskripsi
1	SUB_DBMODULE	Berisi procedure-procedure dan fungsi-fungsi utama program
2	SUB_DBMODULE_INVOICE	Berisi kelas-kelas untuk proses pembayaran

		transaksi
3	SUB_DBMODULE_JOURNAL	Berisi kelas-kelas untuk pencatatan aktifitas user
4	SUB_DBMODULE_LEASING	Berisi kelas-kelas untuk proses pemakaian kamar
5	SUB_DBMODULE_OBJECT	Berisi kelas-kelas yang berhubungan dengan obat-obatan dan tindakan-tindakan
6	SUB_DBMODULE_OWNER	Berisi kelas-kelas yang berhubungan dengan orang-orang yang terlibat dalam sistem
7	SUB_DBMODULE_PROGRAM	Berisi procedure-procedure dan fungsi-fungsi utama program
8	SUB_DBMODULE_TRANSACT	Berisi kelas-kelas yang berhubungan dengan proses transaksi
9	SUB_DBMODULE_MEDICAL	Berisi kelas-kelas yang berhubungan dengan rekam medik pasien

5.1.1 SUB_DBMODULE

DBModule ini berisi procedure-procedure utama program khususnya untuk melakukan inisialisasi program dan koneksi ke database, antara lain :

- 1. Contoh sebagian fungsi inisialisasi dalam dbmodule diatas adalah seperti dibawah ini :

```
function Initiali:ePostgreSQL:Boolean;
var ODBCDriver : String;
begin
  Result:=False;
  ODBCDriver:=GlobalData.WinSystemDir+'\'+'psqlodbc.dll';
  if FileExists(ODBCDriver)=False then begin
    If CopyFile(PChar(GlobalData.Path+'\'+'psqlodbc.dll'),PChar(GlobalData.WinSystemDir+
      '\'+psqlodbc.dll'),False)=False then Exit;
    If FileExists(ODBCDriver)=False then Exit;
  end;
  // Windows ODBC Driver Setup
  with FRegistry do begin
    RootKey:=HKEY_LOCAL_MACHINE;
    if OpenKey('\SOFTWARE\ODBC\ODBCINST.INI\PostgreSQL',False)=False then begin
      // Fail Initalize Registry
      If OpenKey('\SOFTWARE\ODBC\ODBCINST.INI\PostgreSQL',True)=False then Exit;
      WriteString('Setup',ODBCDriver);
      WriteString('Driver',ODBCDriver);
      WriteString('ConnectFunctions','YYN');
      WriteString('DriverODBCVer','02.00');
      WriteString('APILevel','1');
      WriteString('$QLLevel','1');
      WriteString('FileUsage','0');
      WriteString('FileExtns','*.txt');
      WriteInteger('UsageCount',00000001);
      CloseKey;
      OpenKey('\SOFTWARE\OLE32\ODBCINST.INI\ODBC Drivers',True);
      WriteString('PostgreSQL','Installed');
      CloseKey;
    end;
  end;
  Result:=True;
End;
```

Fungsi diatas digunakan untuk inisialisasi pembuatan ODBC PostgreSQL secara langsung ke dalam Windows Registry.

2. Contoh sebagian fungsi untuk koneksi ke database dalam dbmodule diatas adalah seperti dibawah ini :

```
function ConnectPostgreSQL(VServerID,VPort,VDatabaseID,VUserID,VPasswordID:String):Boolean;
var ODBCDriver : String; FPort:String;
begin
  if InitializePostgreSQL=False then Halt;
  ODBCDriver:=GlobalData.WinSystemDir+'\'+'psqlodbc.dll';
  If IsNotNull(VPort) then FPort:=VPort else FPort:='5432';
  with FRegistry do begin
    RootKey:=HKEY_CURRENT_USER;
    OpenKey('\Software\ODBC\ODBC.INI'+ODBCDataLink, True);
    WriteString('Description','PostgreSQL');
    WriteString('Driver',ODBCDriver);
    WriteString('Database',VDatabaseID);
    WriteString('ServerName',VServerID);
    WriteString('Port',FPort);
    WriteString('UserName',VUserID);
    WriteString('Password',VPasswordID);
    WriteString('ReadOnly','0');
    WriteString('ShowOidColumn','0');
    WriteString('FakeOidIndex','0');
    WriteString('RowVersioning','0');
    WriteString('ShowSystemTables','0');
    WriteString('Protocol','0');
    WriteString('ConnSettings','0');
    OpenKey('\Software\ODBC\ODBC.INI\ODBC Data Sources', True);
    WriteString(ODBCDataLink, 'PostgreSQL');
    CloseKey;
  end;
  // Windows ADO Setup
  with dbmodule.ADOConnection do begin
    Close;
    KeepConnection:= True;
    ConnectionString:= 'Provider=MSDASQL.1;Persist Security Info=False, Data Source='+
      ODBCDataLink;
    Try Open(VUserID,VPasswordID);
    except
    end;
    Result:= Connected;
    if Result then begin
      With FRegistry do begin
        RootKey:=HKEY_CURRENT_USER;
        DeleteKey('\Software\ODBC\ODBC.INI'+ODBCDataLink);
        OpenKey('\Software\ODBC\ODBC.INI\ODBC Data Sources', True);
        DeleteValue(ODBCDataLink);
        CloseKey;
      end;
    end;
  end;
End;
```

Fungsi diatas adalah untuk mengkoneksi program ke database database PostgreSQL menggunakan ODBC melalui ADOConnection dimana

ADOConnection ini yang akan dipakai oleh dbmodule-dbmodule lain untuk menjalankan perintah-perintah SQL-nya.

Selain fungsi-fungsi untuk koneksi diatas juga terdapat fungsi-fungsi lain yang berhubungan dengan pembuatan report, seperti dibawah ini :

```
function SayValue(x:integer):string;
const
  FSayNumber : array[0..11] of string[15]=( '','Satu','Dua','Tiga','Empat','Lima',
                                             'Enam','Tujuh','Delapan','Sembilan','Sepuluh','Sebelas');
begin
  if (x<12) then SayValue:=''+FSayNumber[x]
  else If (x<20) then SayValue:=SayValue(x-10)+' Belas'
  else If (x<100) then SayValue:=SayValue(x div 10) + ' Puluh'+SayValue(x mod 10)
  else If (x<200) then SayValue:=' Seratus'+SayValue(x-100)
  else If (x<1000) then SayValue:=SayValue(x div 100)+' Ratus'+SayValue(x mod 100)
  else If (x<2000) then SayValue:=' Seribu'+SayValue(x-1000)
  else If (x<1000000) then SayValue:=SayValue(x div 1000)+' Ribu'+SayValue(x mod 1000)
  else If (x<1000000000) then SayValue:=SayValue(x div 1000000) + ' Juta'+SayValue(x mod 1000000)
  else If (x>=1000000000) then SayValue:=SayValue(x div 1000000000) + ' Miliar'+SayValue(x mod 1000000000);
End;
```

Fungsi diatas dipanggil ketika pembuat kwitansi-kwitansi untuk menghasilkan nilai uang dengan huruf, misal : 10000 menjadi Sepuluh Ribu.

5.1.2 SUB_DBMODULE_INVOICE

DBModule ini berisi class-class utama program khususnya yang berhubungan dengan proses pembayaran pasien rumah sakit. Adapun class-class yang terdapat dalam dbmodule ini adalah :

- **TJenisPembayaran**

Class ini berhubungan dengan proses pengolahan data di tabel JenisPembayaran, adapun deskripsinya adalah seperti dibawah ini :

```
TJenisPembayaran=class
  Sharedata: TJenisPembayaranRecord;
  function Open: boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TJenisPembayaranRecord;
  function SelectJenisPembayaran(VJenisPembayaranID: string): TJenisPembayaranRecord;
end;
```

- **TPembayaran**

Class ini berhubungan dengan proses pengolahan data di tabel

Pembayaran, adapun deskripsinya adalah seperti dibawah ini :

```
TPembayaran=class
  Sharedata: TPembayaranRecord;
  function Open: boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TPembayaranRecord;
  function SelectPembayaran(VPembayaranID: string): TPembayaranRecord;
  function OpenPembayaran(VJenisPembayaranID,VDeptID,VTipePasienID:string): boolean;
end;
```

- **TTransaksiPembayaran**

Class ini berhubungan dengan proses pengolahan data di tabel tabel-tabel yang berhubungan dengan pembayaran pasien, adapun deskripsinya adalah seperti dibawah ini :

```
TTransaksiPembayaran=class
  ShareData: TTransaksiPembayaranRecord;
  function Insert(VTipe:integer): boolean;
  function Update: boolean;
  function Delete(VDPRI_ID:string;VTipe:integer): boolean;
  function Load(VID:string;VTipe:integer): string;
end;
```

Ketika aplikasi memanggil fungsi Insert dalam class ini, aplikasi akan menjalankan fungsi inserttpbyr yang terdapat dalam database, dengan perintah dibawah ini :

```
CREATE OR REPLACE FUNCTION inserttpbyr(VARCHAR,DATE,TIME,VARCHAR,INTEGER)
RETURNS VARCHAR AS '
DECLARE
  v_tpbyrid ALIAS FOR $1;
  v_date ALIAS FOR $2;
  v_time ALIAS FOR $3;
  v_dp ALIAS FOR $4;
  v_tipe ALIAS FOR $5;
  v_tpbyrbr VARCHAR;
  v_id VARCHAR;
BEGIN
  LOCK TABLE TransaksiPbyr IN EXCLUSIVE MODE;
  IF v_tipe = 0 THEN
    LOCK TABLE TPbyrRJ IN EXCLUSIVE MODE;
  END IF;
  IF v_tipe = 1 THEN
    LOCK TABLE TPbyrRI IN EXCLUSIVE MODE;
  END IF;
  IF v_tipe = 2 THEN
    LOCK TABLE TPbyrApt IN EXCLUSIVE MODE;
  END IF;
  SELECT selectsqlrecid(v_tpbyrid) INTO v_id;
  v_tpbyrbr=v_tpbyrid||'/'||v_id;
```



```

INSERT INTO TransaksiPbyr(TPbyr_ID,Tanggal,Jam)
VALUES(v_tpbyrbr,v_date,v_time);
IF v_tipe = 0 THEN
  INSERT INTO TPbyrRJ(TPbyr_ID,PRJ_ID)
  VALUES(v_tpbyrbr,v_dp);
END IF;
IF v_tipe = 1 THEN
  INSERT INTO TPbyrRI(TPbyr_ID,PRI_ID)
  VALUES(v_tpbyrbr,v_dp);
END IF;
IF v_tipe = 2 THEN
  INSERT INTO TPbyrApt(TPbyr_ID,TO_ID)
  VALUES(v_tpbyrbr,v_dp);
END IF;
RETURN v_tpbyrbr;
END;
' LANGUAGE 'plpgsql';

```

- **THutang**

Class ini berhubungan dengan proses pengolahan data di tabel Hutang,

adapun deskripsinya adalah seperti dibawah ini :

```

THutang=class
  Sharedata: THutangRecord;
  TotalHutang: double;
  TotalPembayaranHutang: double;
  function Open(VIPasienID:string): boolean;
  function Insert: boolean;
  function Delete(VHutangID:string): boolean;
  function SelectNewHutang: string;
  function SelectHutang(VHutangID:string): THutangRecord;
  function SelectRecord(VADOQuery: TADOQuery): THutangRecord;
  function UpdateHutangTerbayar(VHutangID:string;VBayar:double): boolean;
  function SelectTotalHutang: boolean;
end;

```

Ketika aplikasi memanggil fungsi Open dalam class ini, aplikasi akan

memanggil view Hutang_v yang terdapat dalam database, dengan perintah

dibawah ini :

```

CREATE VIEW Hutang_v AS
SELECT A.*,D.PasienID,E.PNama,E.PAlamat,'Rawat Jalan' As Deskripsi
FROM Hutang A,TransaksiPbyr B,TPbyrRJ C,PasienRJ D,Pasien E
WHERE ((A.TPbyr_ID=B.TPbyr_ID) AND (B.TPbyr_ID=C.TPbyr_ID) AND
(C.PRJ_ID=D.PRJ_ID) AND (D.PasienID=E.PID))
UNION
SELECT A.*,D.PasienID,E.PNama,E.PAlamat,'Rawat Inap' As Deskripsi
FROM Hutang A,TransaksiPbyr B,TPbyrRI C,PasienRI D,Pasien E
WHERE ((A.TPbyr_ID=B.TPbyr_ID) AND (B.TPbyr_ID=C.TPbyr_ID) AND
(C.PRI_ID=D.PRI_ID) AND (D.PasienID=E.PID))
UNION
SELECT A.*,D.PasienID,E.PNama,E.PAlamat,'Pembelian Obat' As Deskripsi
FROM Hutang A,TransaksiPbyr B,TPbyrApt C,TOPasien D,Pasien E
WHERE ((A.TPbyr_ID=B.TPbyr_ID) AND (B.TPbyr_ID=C.TPbyr_ID) AND
(C.TO_ID=D.TO_ID) AND (D.PasienID=E.PID));

GRANT ALL PRIVILEGES ON Hutang_v TO PUBLIC;

```

- **TPembayaranHutang**

Class ini berhubungan dengan proses pembayaran hutang pasien, adapun deskripsinya adalah seperti dibawah ini :

```
TPembayaranHutang=class
  Sharedata: TPembayaranHutangRecord;
  function Load(VPBH_ID:string): boolean;
  function Insert: boolean;
  function Delete(VPembayaranHutangID:string): boolean;
  function SelectNewPembayaranHutang: string;
end;
```

5.1.3 SUB_DBMODULE_JOURNAL

DBModule ini berisi class TAktifitasJurnal yang digunakan untuk melakukan pencatatan aktifitas-aktifitas user ketika menggunakan aplikasi sistem informasi ini. Adapun deskripsi dari class TAktifitasJurnal yang terdapat dalam dbmodule ini adalah :

```
TAktifitasJurnal= class
  Sharedata: TAktifitasJurnalRecord;
  function Open(VADOQuery:TADOQuery;VMenuParentID:string;VFromDate,VFromTime,VToDate,
    VToTime:TDateTime;VStartLimit,VEndLimit:integer): boolean;
  function Insert: string;
  function Delete: boolean;
  function ClearAktifitasJurnal: TAktifitasJurnalRecord;
  function SelectRecord(VADOQuery:TADOQuery): TAktifitasJurnalRecord;
  function SelectAktifitas(VNoJurnal:string): TAktifitasJurnalRecord;
  function SelectItemID(VResetType:TResetPeriode;VStringReference:String;VDate:TDateTime):Integer;
  function SelectNewNoJurnal:string;
end;
```

5.1.4 SUB_DBMODULE_LEASING

DBModule ini berisi class-class program khususnya yang berhubungan dengan proses pemakaian kamar oleh pasien rumah sakit. Adapun class-class yang terdapat dalam dbmodule ini adalah :

- **TKamar**

Class ini berhubungan dengan proses pengolahan data pada tabel Kamar, adapun deskripsinya adalah seperti dibawah ini :

```
TKamar = class
  ShareData: TKamarRecord;
```



```

function Open(VADOQuery:TADOQuery;VIsActive:string;VDepartemenID:string;
VKelasID:string):Boolean;
function Insert: boolean;
function Update: boolean;
function Delete: boolean;
function SetActive(VIDKamar,VIsActive:String):Boolean;
function SelectRecord(VADOQuery:TADOQuery): TKamarRecord;
function SelectKamar(VIDKamar:string): TKamarRecord;
end;

```

- **TPemakaianKamar**

Class ini berhubungan dengan proses pengolahan data di tabel

PemakaianKamar, adapun deskripsinya adalah seperti dibawah ini :

```

TPemakaianKamar = class
  ShareData: TPemakaianKamarRecord;
  ShareData1: TPemakaianKamarRecord;
  VKamar_ID: string;
  VIsCostCount: boolean;
  function Open(VDPRI_ID:string;VADOQuery:TADOQuery):boolean;
  function SelectNewPemakaianKamar: string;
  function Insert(VPemakaianKamarRecord:TPemakaianKamarRecord): boolean;
  function Update(VPemakaianKamarRecord:TPemakaianKamarRecord): boolean;
  function SelectRecord(VADOQuery: TADOQuery): TPemakaianKamarRecord;
  function SelectPemakaianKamar(VDPRI_ID: string): TPemakaianKamarRecord;
  function Delete(VPMKK_ID: string): boolean;
end;

```

5.1.5 SUB_DBMODULE_OBJECT

DBModule ini berisi class-class utama program khususnya yang berhubungan dengan proses pengolahan data tindakan, obat dan kelas perawatan. Adapun class-class yang terdapat dalam dbmodule ini adalah :

- **TTipeTindakan**

Class ini berhubungan dengan proses pengolahan data tabel TipeTindakan,

adapun deskripsinya adalah seperti dibawah ini :

```

TTipeTindakan=class
  Sharedata: TTipeTindakanRecord;
  function Open: boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TTipeTindakanRecord;
  function SelectTipeTindakan(VTTID:string):TTipeTindakanRecord;
end;

```

- **TTindakan**

Class ini berhubungan dengan proses pengolahan data tabel Tindakan dan tabel-tabel yang berhubungan dengan tabel Tindakan, adapun deskripsinya adalah seperti dibawah ini :

```
TTindakan=class
  Sharedata: TTindakanRecord;
  DeptTindakan: TStringList;
  TipePasienTindakan: TStringList;
  KelasTindakan: TStringList;
  function Open: boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TTindakanRecord;
  function InsertDeptTindakan: boolean;
  function SelectDeptTindakan: boolean;
  function InsertKelasTindakan: boolean;
  function SelectKelasTindakan: boolean;
  function InsertTipePasienTindakan: boolean;
  function SelectTipePasienTindakan: boolean;
  function SelectTindakan(VID,VDepartemenID,VKelasID,VTipePasienID:string): TTindakanRecord;
end;
```

- **TTipeObat**

Class ini berhubungan dengan proses pengolahan data tabel TipeObat, adapun deskripsinya adalah seperti dibawah ini :

```
TTipeObat=class
  Sharedata: TTipeObatRecord;
  function Open: boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TTipeObatRecord;
  function SelectTipeObat(VTipeObatID: string): TTipeObatRecord;
end;
```

- **TObat**

Class ini berhubungan dengan proses pengolahan data tabel Obat, adapun deskripsinya adalah seperti dibawah ini :

```
TObat=class
  Sharedata: TObatRecord;
  ObatTipeObat: TStringList;
  function Open: boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function InsertObatTipeObat: boolean;
  function SelectObatTipeObat: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TObatRecord;
  function SelectObat(VIDObat:string): TObatRecord;
end;
```


- **TKelas**

Class ini berhubungan dengan proses pengolahan data tabel Kelas, adapun deskripsinya adalah seperti dibawah ini :

```
TKelas=class
  Sharedata: TKelasRecord;
  function Open: boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord: TKelasRecord;
end;
```

5.1.6 SUB_DBMODULE_OWNER

DBModule ini berisi class-class utama program khususnya yang berhubungan dengan proses pengolahan data tindakan, obat dan kelas perawatan. Adapun class-class yang terdapat dalam dbmodule ini adalah :

- **TPasien**

Class ini berhubungan dengan proses pengolahan data tabel Pasien, adapun deskripsinya adalah seperti dibawah ini :

```
TPasien=class
  ShareData : TPasienRecord;
  function Open(VADOQuery:TADOQuery;VAll:boolean):boolean;
  function Insert:boolean;
  function Update:boolean;
  function Delete:boolean;
  function SelectRecord(VADOQuery:TADOQuery):TPasienRecord;
  function SetActive(VIsActive:boolean):boolean;
  function SelectCreateDate(VPasienID:string): TDateTime;
  function SelectPasien(VPasienID:string): TPasienRecord;
end;
```

- **TDesiase**

Class ini berhubungan dengan proses pengolahan data tabel ICD dan ICM, adapun deskripsinya adalah seperti dibawah ini :

```
TDisease = class
  ShareData : TDesiaseRecord;
  DeptICD: TStringList;
  function SelectRecord(VADOQuery:TADOQuery;VTableName:String):TDesiaseRecord;
  function Open(VADOQuery:TADOQuery;VDepartemenID,VTableName,
    VFieldName:String):Boolean;
  function Insert(VTableName:String):Boolean;
  function Update(VTableName:String):Boolean;
  function Delete(VTableName:String):Boolean;
```

```
function InsertDeptICD(VTableName:String):boolean;
function SelectDeptICD(VTableName:String):boolean;
End;
```

- **TDepartemen**

Class ini berhubungan dengan proses pengolahan data tabel Departemen, adapun deskripsinya adalah seperti dibawah ini :

```
TDepartemen= class
  ShareData: TDepartemenRecord;
  function Open(VADOQuery:TADOQuery;VAll:boolean;VTipe:integer): boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TDepartemenRecord;
  function SelectDescription(VDID: string): string;
end;
```

- **TDokter**

Class ini berhubungan dengan proses pengolahan data tabel Dokter dan yang berhubungan dengannya, adapun deskripsinya adalah seperti dibawah ini :

```
TDokter= class
  ShareData: TDokterRecord;
  DeptDokterList: TStringList;
  function Open(VADOQuery: TADOQuery;VAll: boolean): boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SetActive(VIsActive: boolean): boolean;
  function SelectRecord(VADOQuery: TADOQuery): TDokterRecord;
  function SelectDokter(VDokterID:string): TDokterRecord;
  function InsertDeptDokter:boolean;
  function SelectDeptDokter:boolean;
end;
```

- **TStaff**

Class ini berhubungan dengan proses pengolahan data tabel Staff dan yang berhubungan dengannya, adapun deskripsinya adalah seperti dibawah ini :

```
TStaff= class
  ShareData: TStaffRecord;
  DeptStaffList: TStringList;
  function Open(VADOQuery: TADOQuery;VAll: boolean): boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SetActive(VIsActive: boolean): boolean;
  function SelectRecord(VADOQuery: TADOQuery): TStaffRecord;
  function SelectStaff(VStaffID:string): TStaffRecord;
  function InsertDeptStaff:boolean;
```



```
function SelectDeptStaff: boolean;
end;
```

- **TDeptDokter**

Class ini berhubungan dengan proses pengolahan data tabel DeptDokter,

adapun deskripsinya adalah seperti dibawah ini :

```
TDeptDokter= class
  ShareData: TDeptDokterRecord;
  function Open(VADOQuery: TADOQuery; VDepartemenID: string): boolean;
  function SelectRecord(VADOQuery: TADOQuery): TDeptDokterRecord;
  function IsExist(VDokterID, VDepartemenID: string): boolean;
  function SelectDokter(VDokterID, VDepartemenID: string): TDeptDokterRecord;
end;
```

- **TDeptStaff**

Class ini berhubungan dengan proses pengolahan data tabel DeptStaff,

adapun deskripsinya adalah seperti dibawah ini :

```
TDeptStaff= class
  ShareData: TDeptStaffRecord;
  function Open(VADOQuery: TADOQuery; VDepartemenID: string): boolean;
  function SelectRecord(VADOQuery: TADOQuery): TDeptStaffRecord;
  function IsExist(VStaffID, VDepartemenID: string): boolean;
  function SelectStaff(VStaffID, VDepartemenID: string): TDeptStaffRecord;
end;
```

- **TTipePasien**

Class ini berhubungan dengan proses pengolahan data tabel TipePasien,

adapun deskripsinya adalah seperti dibawah ini :

```
TTipePasien= class
  ShareData: TTipePasienRecord;
  function Open(VADOQuery: TADOQuery; VAll: boolean): boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SetActive(VIsActive: boolean): boolean;
  function SelectRecord(VADOQuery: TADOQuery): TTipePasienRecord;
  function SelectTipeBayar(VTipePasienID: String): string;
end;
```

- **TTipeKunjungan**

Class ini berhubungan dengan proses pengolahan data tabel

TipeKunjungan, adapun deskripsinya adalah seperti dibawah ini :

```
TTipeKunjungan= class
  ShareData: TTipeKunjunganRecord;
```

```

function Open(VADOQuery:TADOQuery;VAll:boolean): boolean;
function Insert: boolean;
function Update: boolean;
function Delete: boolean;
function SetActive(VIsActive: boolean): boolean;
function SelectRecord(VADOQuery: TADOQuery): TTipeKunjunganRecord;
end;

```

- **TKarcis**

Class ini berhubungan dengan proses pengolahan data tabel Karcis, adapun deskripsinya adalah seperti dibawah ini :

```

TKarcis= class
  ShareData: TKarcisRecord;
  function Open(VADOQuery:TADOQuery;VDepartemenID,VTipeKunjunganID,
    VTipePasienID:string): boolean;
  function SelectKarcisOpen(VADOQuery:TADOQuery;VDepartemenID,VTipeKunjunganID,
    VTipePasienID:string): boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TKarcisRecord;
  function SelectKarcis(VKarcisID: string): TKarcisRecord;
end;

```

- **TDaftarPasienRJ**

Class ini berhubungan dengan proses pengolahan data tabel PasienRJ, adapun deskripsinya adalah seperti dibawah ini :

```

TDaftarPasienRJ= class
  ShareData : TDaftarPasienRJRecord;
  function Open(VADOQuery:TADOQuery;VRegisterDate:TDateTime;VDepartemenID:string;
    VJenisRawat:integer):boolean;
  function Insert:boolean;
  function Update: boolean;
  function Delete(VRegisterDate:TDateTime;VPasienID,VDepartemenID:string): boolean;
  function IsExist(VRegisterDate:TDateTime;VPasienID,VDepartemenID:string): boolean;
  function SelectRecord(VADOQuery:TADOQuery): TDaftarPasienRJRecord;
  function UpdateKeterangan: boolean;
  function SetStatusBayar(VRegisterDate,VDate,VTime:TDateTime;VPasienID,VDepartemenID,
    VStatusBayar:string): boolean;
  function SelectPasien(VPasienID,VDepartemenID:string;VRegisterDate:TDateTime):
    TDaftarPasienRJRecord;
  function SelectNewDPRJ_ID:string;
end;

```

Ketika menjalankan fungsi open dalam class ini, fungsi akan memanggil view DaftarPasienRJ dalam database, adapun perintah viewnya seperti dibawah ini :

```

CREATE VIEW DaftarPasienRJ_v AS
SELECT A.PRJ_ID,A.PasienID,A.DepartemenID,A.AsalDepartemen,A.PasienTipeID,A.DokterID,
  A.StaffID,A.KarcisID,A.TipeKunjunganID,A.RegisterDate,A.RegisterTime,
  A.ServiceDate,A.ServiceTime,A.JenisKunjungan,A.Umur,A.Referal_In,A.Referal_Out,

```



```

A.Pengirim,A.DIn,A.In_Condition,A.Out_Condition,A.StatusBayar,
ROUND((A.RegisterDate-B.PTglLahir)/365) AS Age,B.PNama AS NamaPasien,B.PJKL,
C.DNama AS NamaDepartemen,C.DTipe,D.TNama AS>NamaTipePasien,B.PAlamat,
B.PRT,B.PRW,B.PKelurahan,B.PKecamatan,B.PKota,
E.DNama||", "||E.DSpesialis AS>NamaDokter, F.SNama AS>NamaStaff
FROM PasienRJ A LEFT JOIN Dokter E ON (A.DokterID = E.DID) LEFT JOIN Staff F ON
(A.StaffID = F.SID), Pasien B,Departemen C,TipePasien D
WHERE A.PasienID=B.PID AND A.DepartemenID=C.DID AND A.PasienTipeID=D.TID;

GRANT ALL PRIVILEGES ON DaftarPasienRJ_v TO PUBLIC;

```

Ketika menjalankan fungsi insert dalam class ini, fungsi akan memanggil

fungsi insertpasienrj di database dengan perintah seperti dibawah ini :

```

CREATE OR REPLACE FUNCTION insertpasienrj(VARCHAR,VARCHAR,VARCHAR,
VARCHAR,VARCHAR,VARCHAR,VARCHAR,DATE,TIME,VARCHAR,REAL,
VARCHAR,VARCHAR,VARCHAR,VARCHAR,VARCHAR) RETURNS BOOL AS '
DECLARE
v_dpj AS v_dpid ALIAS FOR $1;
v_pasienid ALIAS FOR $2;
v_deptid ALIAS FOR $3;
v_asldept ALIAS FOR $4;
v_pasientpeid ALIAS FOR $5;
v_karcisid ALIAS FOR $6;
v_tkid ALIAS FOR $7;
v_regdate ALIAS FOR $8;
v_regtime ALIAS FOR $9;
v_jk ALIAS FOR $10;
v_umur ALIAS FOR $11;
v_refin ALIAS FOR $12;
v_pengirim ALIAS FOR $13;
v_dxin ALIAS FOR $14;
v_incond ALIAS FOR $15;
v_bayar ALIAS FOR $16;
v_jumawal INTEGER;
v_jumakhir INTEGER;
BEGIN
LOCK TABLE PasienRJ IN EXCLUSIVE MODE;
SELECT Count(PRJ_ID) INTO v_jumawal FROM PasienRJ;
INSERT INTO PasienRJ(PRJ_ID,PasienID,DepartemenID,AsalDepartemen,PasienTipeID,
KarcisID,TipeKunjunganID,RegisterDate,RegisterTime,JenisKunjungan,Umur,
Referral_In,Pengirim,DIn, In_Condition,StatusBayar)
VALUES(v_dpj,v_pasienid,v_deptid,v_asldept,v_pasientpeid,v_karcisid,v_tkid,
v_regdate,v_regtime,v_jk,v_umur,v_refin,v_pengirim,v_dxin,v_inCond,v_bayar);
SELECT Count(PRJ_ID) INTO v_jumakhir FROM PasienRJ;
if v_jumakhir > v_jumawal then
return true;
else return false;
end if;
END;
' LANGUAGE 'plpgsql';

```

- **TDiagnosaRJ**

Class ini berhubungan dengan proses pengolahan data tabel DiagnosaRJ,

adapun deskripsinya adalah seperti dibawah ini :

```

TDiagnosaRJ= class
Sharedata: TDiagnosaRecord;
function Open(VADOQuery: TADOQuery; VDPRJ_ID:string): boolean;
function Insert: boolean;

```

```

function Update: boolean;
function Delete: boolean;
function SelectRecord(VADOQuery: TADOQuery): TDiagnosaRecord;
function SelectNewDiagnosaRJ: string;
end;

```

- **TTerapiRJ**

Class ini berhubungan dengan proses pengolahan data tabel TerapiRJ,

adapun deskripsinya adalah seperti dibawah ini :

```

TTerapiRJ= class
  Sharedata: TTerapiRecord;
  function Open(VADOQuery: TADOQuery; VDPRI_ID:string): boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TTerapiRecord;
  function SelectNewTerapiRJ: string;
end;

```

- **TDaftarPasienRI**

Class ini berhubungan dengan proses pengolahan data tabel PasienRI,

adapun deskripsinya adalah seperti dibawah ini :

```

TDaftarPasienRI= class
  ShareData : TDaftarPasienRIRecord;
  function Open(VADOQuery:TADOQuery;VDepartemenID,VIsAktif:string;VStartDate,
    VEndDate:TDateTime;VIsUseDate:boolean):boolean;
  function Insert:boolean;
  function Update:boolean;
  function IsExist(VPasienID:string): boolean;
  function SelectRecord(VADOQuery:TADOQuery): TDaftarPasienRIRecord;
  function UpdateKamar(VDPRI_ID:string;VIDKamar,VDepartemenID:string): boolean;
  function UpdateKeterangan: boolean;
  function SelectPasien(VPasienID,VDepartemenID,VIsAktif:string): TDaftarPasienRIRecord;
  function SelectDPRI(VDPRI_ID:string): TDaftarPasienRIRecord;
  function Delete(VDPRI_ID:string): boolean;
  function KeluarRS(VDPRI_ID:string;VEndDate,VEndTime:TDateTime;
    VUangMuka:double): boolean;
  function BatallKeluarRS(VDPRI_ID:string): boolean;
  function SelectNewDPRI_ID:string;
end;

```

Ketika menjalankan fungsi open dalam class ini, fungsi akan memanggil

view DaftarPasienRJ dalam database, adapun perintah viewnya seperti

dibawah ini :

```

CREATE VIEW DaftarPasienRI_v AS
SELECT A.*, ROUND((CURRENT_DATE-B.PTglLahir)/365) AS Age,B.PJKL,
      B.PNama AS NamaPasien,C.DNama AS NamaDepartemen,D.TNama AS
NamaTipePasien,
      B.PAlamat,B.PRT,B.PRW,B.PKelurahan,B.PKecamatan,B.PFoto,
      L.DNama||", "||E.DSpesialis AS NamaDokter,F.Kamar_Nama,H.Kelas_ID
FROM PasienRI A LEFT JOIN Dokter E ON (A.DokterID = E.DID) LEFT JOIN Kamar F

```



```
ON (A.Kamar_ID = F.Kamar_ID), Pasien B,Departemen C,TipePasien D
WHERE A.PasienID=B.PID AND A.DepartemenID=C.DID AND A.PasienTipeID=D.TID;
GRANT ALL PRIVILEGES ON DaftarPasienRI_v TO PUBLIC;
```

- **TDiagnosaRI**

Class ini berhubungan dengan proses pengolahan data tabel DiagnosaRI, adapun deskripsinya adalah seperti dibawah ini :

```
TDiagnosaRI= class
  Sharedata: TDiagnosaRecord;
  function Open(VADOQuery: TADOQuery; VDPRI_ID:string): boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TDiagnosaRecord;
  function SelectNewDiagnosaRI: string;
end;
```

- **TTerapiRI**

Class ini berhubungan dengan proses pengolahan data tabel TerapiRI, adapun deskripsinya adalah seperti dibawah ini :

```
TKelas=class
  Sharedata: TKelasRecord;
  function Open: boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord: TKelasRecord;
end;
```

- **TDeskripsi**

Class ini berhubungan dengan proses pengolahan data tabel Deskripsi, adapun deskripsinya adalah seperti dibawah ini :

```
TDeskripsi= class
  ShareData: TDeskripsiRecord;
  function Open(VADOQuery: TADOQuery;VDepartemenTipe,VType:Integer): boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery: TADOQuery): TDeskripsiRecord;
end;
```

5.1.7 SUB_DBMODULE_PROGRAM

DBModule ini berisi procedure-procedure utama program khususnya untuk melakukan pengolahan data user dan menu program, antara lain :

1. Contoh procedure untuk mengambil data menu user dari database adalah seperti dibawah ini :

```

procedure LoadUserMenuPrivilege(VUserID:Integer);
var Find : Boolean;
    FString: string;
begin
    isSQLError:=False;
    with dbModuleProgram, dbModule.QCommonQuery do begin
        FString:= 'SELECT '+
            ' A.ActiveStatus, A.LevelID, A.ReclD, A.ParentID, A.Mnemonic, '+
            ' A.DepartmentID, A.DepartmentName, A.MenuStatus, A.MenuDescription, '+
            ' A.ObjectType, A.ObjectName, A.ObjectParam, '+
            ' B.ShowState, B.Tambah, B.Ubah, B.Hapus '+
            ' FROM '+
            ' UserModule A LEFT JOIN UserPrivilege B ON (( A.ReclD=B.ReclD ) AND
            ' ( B.UserID=IntToStr(VUserID)+ ' )) '+
            ' ORDER BY A.ReclD ';
        Close;
        SQL.Clear;
        SQL.Add(FString);
        Try Open;
        Except On EDBEngineError Do dbModule.DBErrorHandling('UserModule',0,0);
        End;
        if DataMenu.Active then DataMenu.EmptyTable else DataMenu.Open;
        while EOF=False do begin
            // ##### //
            DataMenu.Append;
            DataMenu.Fields[0].AsString:=Fields[0].AsString;
            DataMenu.Fields[1].AsInteger:=Fields[1].AsInteger;
            DataMenu.Fields[2].AsInteger:=Fields[2].AsInteger;
            DataMenu.Fields[3].AsInteger:=Fields[3].AsInteger;
            DataMenu.Fields[4].AsString:=Fields[4].AsString;
            DataMenu.Fields[5].AsString:=Fields[5].AsString;
            DataMenu.Fields[6].AsString:=Fields[6].AsString;
            DataMenu.Fields[7].AsInteger:=Fields[7].AsInteger;
            DataMenu.Fields[8].AsString:=Fields[8].AsString;
            DataMenu.Fields[9].AsInteger:=Fields[9].AsInteger;
            DataMenu.Fields[10].AsString:=Fields[10].AsString;
            DataMenu.Fields[11].AsString:=Fields[11].AsString;
            DataMenu.Fields[12].AsInteger:=Fields[12].AsInteger;
            DataMenu.Fields[13].AsString:=Fields[13].AsString;
            DataMenu.Fields[14].AsString:=Fields[14].AsString;
            DataMenu.Fields[15].AsString:=Fields[15].AsString;
            DataMenu.Post;
            // ##### //
            Next;
        end;
        Close;
    end;
end;

```

Fungsi diatas digunakan untuk mengambil data menu dari tabel UserModule di JOIN-kan dengan tabel UserPrivilege yang dimiliki oleh user kemudian hasilnya dimasukkan ke dalam Memory Tabel yang digunakan untuk menu utama aplikasi ini.

2. Contoh procedure untuk mengambil user dari tabel UserAccess adalah seperti dibawah ini :

```
function SelectUserAccess(VUserLogin:String):TUserRecord;
var FString: string;
begin
  with dbModuleProgram.QCommonQuery do begin
    FString:='SELECT
UserID,GroupID,HostID,LevelID,UserLoginID,UserPassID,OwnurID,OwnerName '+
'FROM UserAccess '+
'WHERE (UserLoginID='+SQLString(UPPERCASE(VUserLogin))+')';
    Close;
    SQL.Clear;
    SQL.Add(FString);
    try Open;
    except On EDBEngineError Do dbModule.DBErrorHandling('UserAccess',0,0);
    end;
    Result:=SelectUserRecord(dbModuleProgram.QCommonQuery);
    Close;
  end;
end;
```

Fungsi diatas digunakan untuk mengambil data user tabel UserAccess yang dimasukkan ke record TUserRecord.

5.1.8 SUB_DBMODULE_TRANSACT

DBModule ini berisi class-class utama program khususnya yang berhubungan dengan proses pengolahan data transaksi pasien seperti data transaksi obat, data transaksi tindakan, dan data transaksi lainnya. Adapun class-class yang terdapat dalam dbmodule ini adalah :

- **TTransaksiTindakan**

Class ini berhubungan dengan proses pengolahan data tabel-tabel yang berhubungan dengan transaksi tindakan pasien, adapun deskripsinya adalah seperti dibawah ini :

```
TTransaksiTindakan=class
  ShareData: TTransaksiTindakanRecord;
  function Insert(VType:integer): boolean;
  function Update: boolean;
  function Delete: boolean;
  function Load(VIDP_ID:string;VType:integer): string;
  function LoadDetail(VTT_ID:string): string;
  function InsertTindakanStaff(VTT_Dtl,VStaffID:string;VType:integer): boolean;
  function InsertTindakanDokter(VTT_Dtl,VDokterID:string;VType:integer): boolean;
  procedure FilterRecord(DataSet: TDataSet; var Accept: Boolean);
```

end;

Ketika menjalankan fungsi insert dalam class ini, fungsi akan memanggil fungsi inserttransaksitx di database dengan perintah seperti dibawah ini :

```
CREATE OR REPLACE FUNCTION inserttransaksitx(VARCHAR,VARCHAR,VARCHAR,DATE,
      TIME,VARCHAR,INTEGER) RETURNS VARCHAR AS '
DECLARE
  v_tt看id ALIAS FOR $1;
  v_tpasienid ALIAS FOR $2;
  v_deptid ALIAS FOR $3;
  v_date ALIAS FOR $4;
  v_time ALIAS FOR $5;
  v_dp ALIAS FOR $6;
  v_tipe ALIAS FOR $7;
  v_ttbaru VARCHAR;
  v_id VARCHAR;
BEGIN
  LOCK TABLE TransaksiTx IN EXCLUSIVE MODE;
  IF v_tipe = 0 THEN
    LOCK TABLE TransaksiTxRJ IN EXCLUSIVE MODE;
  ELSE
    LOCK TABLE TransaksiTxRI IN EXCLUSIVE MODE;
  END IF;
  SELECT selectsqlrecid(v_tt看id) INTO v_id;
  v_ttbaru=v_tt看id||"/"||v_id;
  INSERT INTO TransaksiTx(TT_ID,TPasien_ID,Tanggal,Jam)
  VALUES(v_ttbaru,v_tpasienid,v_date,v_time);
  IF v_tipe = 0 THEN
    INSERT INTO TransaksiTxRJ(TT_ID,PRJ_ID)
    VALUES(v_ttbaru,v_dp);
  ELSE
    INSERT INTO TransaksiTxRI(TT_ID,PRI_ID,Dept_ID)
    VALUES(v_ttbaru,v_dp,v_deptid);
  END IF;
  RETURN v_ttbaru;
END;
' LANGUAGE 'plpgsql';
```

- **TTransaksiObatUmum**

Class ini berhubungan dengan proses pengolahan data tabel yang berhubungan dengan proses pembelian obat rawat jalan, adapun deskripsinya adalah seperti dibawah ini :

```
TTransaksiObatUmum=class
  ShareData: TTransaksiObatRecord;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function LoadDetail(VTO_ID:string): boolean;
end;
```

Ketika menjalankan fungsi insert dalam class ini, fungsi akan memanggil fungsi inserttoumum di database dengan perintah seperti dibawah ini :

```
CREATE OR REPLACE FUNCTION inserttoumum(VARCHAR,VARCHAR,VARCHAR,VARCHAR,
```



```

        VARCHAR,DATE,TIME,VARCHAR,VARCHAR,VARCHAR) RETURNS VARCHAR AS '
DECLARE
v_toid ALIAS FOR $1;
v_pasien ALIAS FOR $2;
v_dokter ALIAS FOR $3;
v_alamat ALIAS FOR $4;
v_jenispl ALIAS FOR $5;
v_tanggal ALIAS FOR $6;
v_jam ALIAS FOR $7;
v_pasienid ALIAS FOR $8;
v_tpasienid ALIAS FOR $9;
v_dokterid ALIAS FOR $10;
v_tobaru VARCHAR;
v_id VARCHAR;
BEGIN
LOCK TABLE TObatUmum IN EXCLUSIVE MODE;
SELECT selectsqlrecid(v_toid) INTO v_id;
v_tobaru=v_toid||"/"||v_id;
INSERT INTO TObatUmum(TO_ID,NamaPasien,NamaDokter,Alamat,JenisPbl_ID,Tanggal,Jam)
VALUES(v_tobaru,v_pasien,v_dokter,v_alamat,v_jenispl,v_tanggal,v_jam);
IF length(v_pasienid)>0 THEN
LOCK TABLE TOPasien IN EXCLUSIVE MODE;
INSERT INTO TOPasien(TO_ID,PasienID,TPasien_ID)
VALUES(v_tobaru,v_pasienid,v_tpasienid);
END IF;
IF length(v_dokterid)>0 THEN
LOCK TABLE TODokter IN EXCLUSIVE MODE;
INSERT INTO TODokter(TO_ID,DokterID)
VALUES(v_tobaru,v_dokterid);
END IF;
RETURN v_tobaru;
END;
' LANGUAGE 'plpgsql';

```

• TTransaksiObatRI

Class ini berhubungan dengan proses pengolahan data tabel-tabel yang berhubungan dengan proses pembelian obat pasien rawat inap, adapun deskripsinya adalah seperti dibawah ini :

```

TTransaksiObatRI=class
ShareData: TTransaksiObatRecord;
function Insert: boolean;
function Update: boolean;
function Delete: boolean;
function LoadDetail(VTORI_ID:string): boolean;
end;

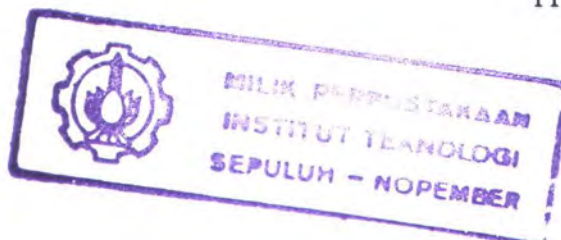
```

Ketika menjalankan fungsi insert dalam class ini, fungsi akan memanggil fungsi inserttori di database dengan perintah seperti dibawah ini :

```

CREATE OR REPLACE FUNCTION inserttori(VARCHAR,VARCHAR,VARCHAR,VARCHAR,
DATE,TIME) RETURNS VARCHAR AS '
DECLARE
v_toriid ALIAS FOR $1;
v_priid ALIAS FOR $2;
v_jenispl ALIAS FOR $3;
v_dokterid ALIAS FOR $4;
v_tanggal ALIAS FOR $5;
v_jam ALIAS FOR $6;

```



```

v_toribaru VARCHAR;
v_id VARCHAR;
BEGIN
LOCK TABLE TObatRI IN EXCLUSIVE MODE;
SELECT selectsqlrecid(v_toriid) INTO v_id;
v_toribaru=v_toriid||"/"||v_id;
INSERT INTO TObatRI(TORI_ID,PRI_ID,JenisPbl_ID,DokterID,Tanggal,Jam)
VALUES(v_toribaru,v_priid,v_jenispbl,v_dokterid,v_tanggal,v_jam);
RETURN v_toribaru;
END;
' LANGUAGE 'plpgsql';

```

• **TRincianTransaksiRI**

Class iri berhubungan dengan proses pengolahan data tabel

RincianTransaksiRI, adapun deskripsinya adalah seperti dibawah ini :

```

TRincianTransaksiRI=class
  Sharedata: TRincianTransaksiRIRecord;
  function ClearRincianTransaksiRI: TRincianTransaksiRIRecord;
  function SelectNewRincianTransaksiRI:string;
  function SelectTindakan(VPRI_ID:string): boolean;
  function SelectObat(VPRI_ID:string;VTipe:integer): double;
  function SelectKamar(VPRI_ID:string;VStopDate,VStopTime:TDateTime): double;
  function Load(VPRI_ID:string): boolean;
  function Insert: boolean;
  function Delete(VPRI_ID:string): boolean;
end;

```

• **TJenisPembelian**

Class ini berhubungan dengan proses pengolahan data tabel

JenisPembelian, adapun deskripsinya adalah seperti dibawah ini :

```

TJenisPembelian=class
  Sharedata: TJenisPembelianRecord;
  SetHarga: TSetHargaJualRecord;
  function Open(VADOQuery:TADOQuery): boolean;
  function Insert: boolean;
  function Update: boolean;
  function Delete: boolean;
  function SelectRecord(VADOQuery:TADOQuery): TJenisPembelianRecord;
  function InsertSetHarga(VJenisPembelianID:string): boolean;
  function DeleteSetHarga(VJenisPembelianID:string): boolean;
  function SelectSetHarga(VJenisPembelianID:string): boolean;
end;

```

5.1.9 SUB_DBMODULE_MEDICAL

DBModule ini berisi class-class utama program khususnya yang berhubungan dengan proses rekam medik pasien. Adapun class-class yang terdapat dalam dbmodule ini adalah :

- **TRekamMedik**

Class ini berhubungan dengan proses pengolahan data tabel-tabel yang berhubungan dengan rekam medik pasien, adapun deskripsinya adalah seperti dibawah ini :

```
TRekamMedik = class
  ShareData : TRekamMedikRecord;
  procedure OpenReportTable(VPasienID,VNamaPasien,VALamat:string;VUmur:integer);
  function Open(VDP_ID:string;VTipe:integer): string;
  function OpenDetail(VRM_ID:string): string;
  function Insert(VTipe:integer): boolean;
  function Delete: boolean;
  function Update: boolean;
end;
```

Ketika menjalankan fungsi open dalam class ini, fungsi akan memanggil view RekamMedik_v di database dengan perintah seperti dibawah ini :

```
CREATE VIEW RekamMedik_v AS
SELECT A.*,B.Tanggal,B.Jam,D.PasienID
FROM RekamMedik_Dtl A, RekamMedik B, RekamMedikRJ C, PasienRJ D
WHERE (A.RM_ID=B.RM_ID) / ND (B.RM_ID=C.RM_ID) AND (C.PRJ_ID=D.PRJ_ID)
UNION
SELECT A.*,B.Tanggal,B.Jam,D.PasienID
FROM RekamMedik_Dtl A, RekamMedik B, RekamMedikRI C, PasienRI D, Pasien E
WHERE (A.RM_ID=B.RM_ID) AND (B.RM_ID=C.RM_ID) AND (C.PRI_ID=D.PRI_ID);
GRANT ALL PRIVILEGES ON RekamMedik_v TO PUBLIC;
```

Ketika menjalankan fungsi insert dalam class ini, fungsi akan memanggil fungsi insertrekammedik di database dengan perintah seperti dibawah ini :

```
CREATE OR REPLACE FUNCTION insertrekammedik(VARCHAR,VARCHAR,DATE,TIME,
  VARCHAR,INTEGER) RETURNS VARCHAR AS '
DECLARE
  v_rmid ALIAS FOR $1;
  v_dokterid ALIAS FOR $2;
  v_date ALIAS FOR $3;
  v_time ALIAS FOR $4;
  v_dp ALIAS FOR $5;
  v_tipe ALIAS FOR $6;
  v_rmbaru VARCHAR;
  v_id VARCHAR;
BEGIN
  LOCK TABLE RekamMedik IN EXCLUSIVE MODE;
  IF v_tipe = 0 THEN
    LOCK TABLE RekamMedikRJ IN EXCLUSIVE MODE;
  ELSE
    LOCK TABLE RekamMedikRI IN EXCLUSIVE MODE;
  END IF;
  SELECT selectsqlrecid(v_rmid) INTO v_id;
  v_rmbaru=v_rmid||"/"||v_id;
  INSERT INTO RekamMedik(RM_ID,Dokter_ID,Tanggal,Jam)
  VALUES(v_rmbaru,v_dokterid,v_date,v_time);
  IF v_tipe = 0 THEN
    INSERT INTO RekamMedikRJ(RM_ID,PRJ_ID)
    VALUES(v_rmbaru,v_dp);
```

```

ELSE
  INSERT INTO RekamMedikRI(RM_ID,PRI_ID)
  VALUES(v_rmbaru,v_dp);
END IF;
RETURN v_rmbaru;
END;
' LANGUAGE 'plpgsql';

```

- **TTemplate**

Class ini berhubungan dengan template rekam medik, adapun deskripsinya adalah seperti dibawah ini :

```

TTemplate=class
  Sharedata: TTemplateRecord;
  ShareRMTemplate: TRMTemplateRecord;
  ResultString: TStringList;
  DeptTemplateList: TStringList;
  function Open(VADOQuery:TADOQuery):boolean;
  function OpenDeptTemplate(VADOQuery:TADOQuery;VKeyWord, VDepartemenID:String):
    boolean;
  function Insert:boolean;
  function Update:boolean;
  function Delete:boolean;
  function SelectRecord(VADOQuery:TADOQuery):TTemplateRecord;
  function SelectRecordRMTemplate:TRMTemplateRecord;
  function OpenRMTemplate(VTemplateID:string):boolean;
  function OpenRMResultString(VRMTemplateID:string):boolean;
  function SelectNewRMTemplate: string;
  function InsertRMTemplate(VTemplateID:string):boolean;
  function InsertRMResultString(VRMTemplateID:string):boolean;
  function DeleteRMTemplate(VTemplateID:string):boolean;
  function InsertDeptTemplate:boolean;
  function SelectDeptTemplate:boolean;
end;

```

5.2 Uji Coba Sistem

Tujuan uji coba adalah untuk mengetahui apakah perangkat lunak yang dihasilkan dapat menjalankan fungsi-fungsinya dengan baik dan sesuai dengan tujuan Tugas Akhir ini. Uji coba dilakukan dalam lingkungan dan skenario yang telah ditentukan.

5.2.1 Lingkungan Uji Coba

Uji coba dilakukan pada komputer yang memiliki spesifikasi prosesor AMD Athlon XP 1.8 GHz dengan memori 256 MB. Sistem operasi yang

dijalankan pada komputer tersebut adalah MS Windows XP. Komputer tersebut digunakan sebagai server database PostgreSQL dan sekaligus menjalankan aplikasi Sistem Informasi Pasien dan Rekam Medik.

5.2.2 Skenario Uji Coba

Skenario yang akan diujikan adalah sebagai berikut :

1. Uji coba pelayanan rawat jalan, mulai pendaftaran pasien, pengisian rekam medik pasien serta pengisian tindakan medik di poliklinik sampai dengan pasien pulang dari rumah sakit.
2. Uji coba pelayanan rawat inap, mulai pendaftaran pasien, pengisian rekam medik pasien serta pengisian tindakan medik, pembelian obat rawat inap sampai dengan kepulangan pasien dari rumah sakit.
3. Uji coba pelayanan apotek umum, dari pengisian obat sampai dengan pengisian pembayaran apotek.
4. Uji coba pelayanan pembayaran hutang pasien.
5. Uji coba pencetakan laporan rumah sakit.

5.2.3 Pelaksanaan Uji Coba

Setelah semua sarana fisik dan perangkat lunak pendukung siap, maka dilakukan uji coba aplikasi yang telah dibuat. Sebelum uji coba dilakukan, dilakukan terlebih dulu penataan beberapa konfigurasi pada server database PostgreSQL agar bisa terkoneksi dengan Aplikasi Sistem Informasi. Serta dilakukan import data-data yang dibutuhkan oleh sistem. Setelah persiapan dilakukan dengan benar, barulah aplikasi siap dijalankan.

5.2.3.1 Pengaturan Konfigurasi server PostgreSQL

Setelah melakukan penginstalan server PostgreSQL baik dari Windows platform ataupun Linux Platform, dilakukan beberapa pengesetan pada file setting dari PostgreSQL. Pada Linux Platform harus dilakukan pengeditan file `postgresql.conf` untuk membuka port yang digunakan pada koneksi server PostgreSQL seperti pada konfigurasi dibawah ini :

```
# - Connection Settings -
listen_addresses = '*'          # what IP interface(s) to listen on;
                                # default's to localhost, '*' = any
port = 5432
max_connections = 100
# note: increasing max_connections costs about 500 bytes of shared
# memory per connection slot, in addition to costs from shared_buffers
# and max_locks_per_transaction.
```

Serta perlu dilakukan pengeditan pada file `pg_hba.conf` untuk mengeset host mana saja yang bisa terkoneksi pada server PostgreSQL.

```
# TYPE DATABASE  USER  CIDR-ADDRESS  METHOD
# IPv4 local connections:
host all all 127.0.0.1/32 md5
host all all 192.168.1.1/128 md5
# IPv6 local connections:
host all all ::1/128 md5
```

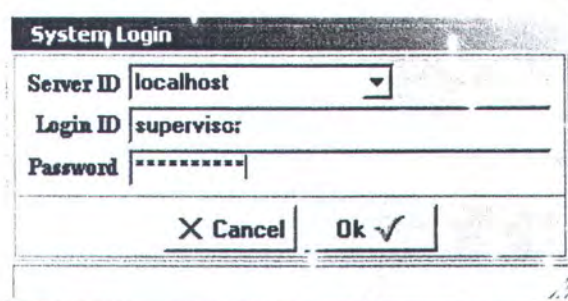
Setelah itu baru membuat user baru supervisor serta database baru `rsi_benowo` dengan perintah

```
CREATE USER supervisor WITH PASSWORD 'supervisor' CREATEDB CREATEUSER
CREATEDB rsi_benowo
```

Untuk penggantian user password supervisor dapat diganti dalam Aplikasi Sistem Informasi.

5.2.3.2 Login User

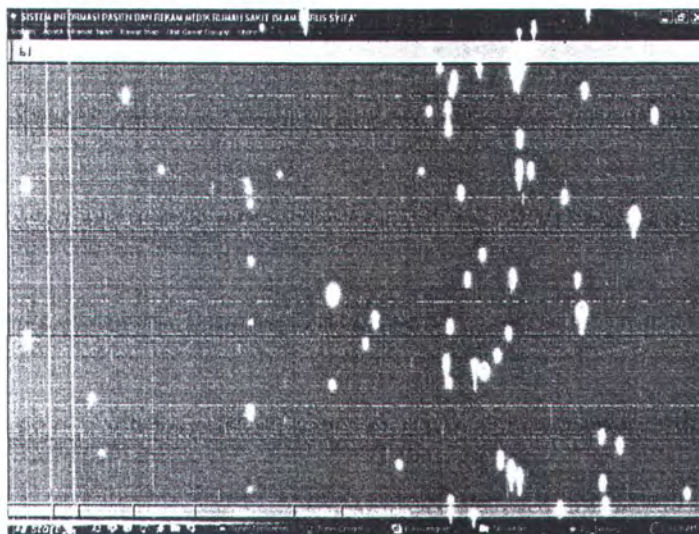
Untuk menjalankan aplikasi, kita menjalankan aplikasi “RSI BENOWO.EXE”. Ketika pertama kali aplikasi dijalankan akan ditampilkan form login user. Berikut ini adalah tampilan form tersebut.



Gambar 5.1 Form Login User

Kemudian kita memasukkan login beserta password yang sesuai dengan user pada aplikasi. Bila login serta password user sesuai, akan ditampilkan menu utama dari aplikasi ini.

Diasumsikan bahwa seluruh menu aplikasi sudah diset, serta sudah dilakukan data-data master rumah sakit seperti pasien, dokter, tindakan, obat dan lain-lain.



Gambar 5.2 Form Main Menu

5.2.3.3 Uji Coba Pelayanan Rawat Jalan

Diasumsikan Pasien Hendro Goenawan dengan Rekam Medik (069636) melakukan pendaftaran ke Poliklinik Gigi.

Untuk melakukan pendaftaran Rawat Jalan kita masuk ke menu Rawat Jalan – Registrasi – Registrasi Rawat Jalan. Kemudian akan tampil form pendaftaran rawat jalan seperti dibawah ini.

Gambar 5.3 Form Daftar Pasien Rawat Jalan

Untuk melakukan pendaftaran pasien lama yang sudah mempunyai nomor rekam medik klik tombol Re untuk menampilkan input pasien rawat jalan pasien lama, seperti dibawah ini :

RM	069636		
Nama	HENDRO GOENAWAN	L	Lahir 26/05/1979 26
Alamat	JL. PERMATA HIJAU A-13		
RT / RW	3 / 9	Desa / Kecamatan	TLOGOMAS
Kota	MALANG	Gol Darah	O
Poli	GIGI	Kategori	UMUM
T.Bayar	PEMERIKSAAN	Karcis	KRC06
		Biaya	Rp 5,000.00
Cara Msk.	Datang Sendiri	Oleh	
DX	CHECKUP HARIAN	Kondisi Msk.	Terlihat Sehat
Batal		Input	Edit Data Pasien

Gambar 5.4 Form Input Pasien Rawat Jalan

Kemudian ketika user memasukkan Nomor Rekam Medik pasien maka aplikasi akan mencari data pasien dalam database. Kemudian user memasukkan poliklinik, tipe pasien, karcis, cara masuk, diagnosa masuk, serta kondisi pasien. Setelah selesai klik input untuk memasukkan ke dalam daftar pasien rawat jalan seperti dibawah ini :

No. RM	Nama	Alamat	S	U	Poliklinik	Kategori
069636	HENDRO GOENAWAN	JL. PERMATA HIJAU A-13	L	25	GIGI	UMUM

Gambar 5.5 Form Pasien Rawat Jalan Updated

Untuk mencetak karcis Rawat Jalan, klik tombol Karcis di pojok kanan bawah, yang akan menghasilkan kwitansi seperti dibawah ini :

RSI Darus Syifa Benowo Jl. Raya Benowo, Surabaya Telpon : (031) 1234435		UMUM
Nama Pasien	: HENDRO GOENAWAN (069636)	No. RM : 069636
Terbilang	: Lima Ribu Rupiah	
Untuk Pembayaran	: Karcis Poliklinik Gigi Pasien UMUM	
Rp 5.000.00		10-01-2005
Jam Datang : 10:14:58		NANA
Jam Cetak : 10:21:27		

Gambar 5.6 Kwitansi Karcis Rawat Jalan

Pasien menuju ke tempat perawatan untuk diperiksa oleh dokter.

Kemudian Perawat akan memanggil pasien yang terdapat di Daftar Pasien Rawat Jalan sesuai dengan urutan dalam daftar yang disesuaikan dengan urutan kedatangan pasien, seperti gambar dibawah ini :

Gambar 5.7 Daftar Pasien Poliklinik Gigi

Dalam Form ini terdapat beberapa fungsi antara lain :

- 1. Memasukkan Keterangan Pasien
Klik tombol Ubah untuk melakukan edit dokter yang merawat, perawat, kondisi data, cara data, kondisi pulang serta cara pulang.
- 2. Memasukkan Diagnosa Pasien
Klik tombol Tambah Dx untuk menambah Diagnosa Pasien.
- 3. Memasukkan Terapi Pasien
Klik tombol Tambah Terapi untuk menambah Terapi Pasien
- 4. Memasukkan Rekam Medik
Klik tombol Rekam Medik kemudian klik tombol Tambah yang akan memunculkan input data Free Form Rekam Medik pasien.

Free Form								X
F1 Subjective	F2 Objective	F3 Laboratory	F4 Assessment	F5 Planning	F6 Therapy	F7 Receipt		
CHECKUP Lagi 1 bulan lagi								
F1 Subj	F2 Obj	F3 Lab	F4 Asst	F5 Plan	F6 Tx	F7 Rx	F10 Save	

Gambar 5.8 Form MedRec Free Form

Tekan F10 untuk menyimpan data rekam medik ke tabel sementara seperti gambar dibawah ini :

REKAM MEDIK PASIEN

File

Nama Pasien

HENDRO GOENAWAN

069636

Nama Dokter

Dwi Hariyanto, drg.

DKT017

Tanggal

10/01/2005

10:41:05 AM

Catatan Medik Pasien

Type	Jenis Pemeriksaan	Deskripsi
Subjective	Free Form	Karies Gigi berkurang
Objective	Free Form	Checkup Karies Gigi
Assesment	Free Form	KARIES GIGI
Planning	Free Form	CHECKUP Lagi 1 bulan lagi

Tambah

Ubah

Hapus

Batal

OK

Simpan

Layout

Tampil

Desain

RM Lama

Keluar

Gambar 5.9 Form Rekam Medik Pasien

Untuk Melihat Rekam Medik lama pasien klik pada tombol RM

Lama, yang akan memunculkan report Rekam Medik lama pasien :

Data Rekam Medik Pasien RSI Darus Syifa Benowo

Nama Pasien : HENDRO GOENAWAN (069636)

Alamat : JL. PERMATA HIJAU A-13

Umur : 26 tahun

Halaman : 1

Tanggal	Jam	Sub jective	Objective	Assesment	Planning
09/01/2005	13:19:14	Baraf	rebaraf	rebaraf	
05/01/2005	11:33:42	rebaraf	rebaraf	rebaraf	rebaraf

Gambar 5.10 Laporan Rekam Medik Pasien

Tekan tombol Simpan untuk memasukkan data ke dalam database.

5. Memasukkan Transaksi Tindakan

Klik tombol Transaksi kemudian klik tombol Tambah yang akan menambah data transaksi yang dilakukan oleh pasien.

TINDAKAN MEDIK PASIEN

Nama Pasien: HENDRO GOENAWAN | 069636 | Tipe Pasien: UMUM

Tanggal: 10/01/2005 | 10:52:43 AM | Departemen:

Tindakan Medik Pasien: Pembayaran

No	ID	Nama	Jml	Harga
1	PG009	Pembayaran Cash Poliklinik Gigi (UMUM)	1	45.000,00

No Item: 1
ID: PG009
Jumlah: 1
Harga: 45.000,00
Total: 45.000,00

Pelaksana Tindakan

ID	Nama Staff	Tipe Staff	Tipe Tr
----	------------	------------	---------

Total Transaksi: 45.000,00

Tambah, Ubah, Hapus, Batal, OK, Simpan, Tampil, Cetak, Desain, Keluar

Gambar 5.11 Form Transaksi Tindakan Pasien

Untuk melakukan Pembayaran transaksi tekan tab dialog pembayaran terus klik tombol Tambah untuk menambah data pembayaran.

TINDAKAN MEDIK PASIEN

Nama Pasien: HENDRO GOENAWAN | 069636 | Tipe Pasien: UMUM

Tanggal: 10/01/2005 | 10:52:43 AM | Departemen:

Tindakan Medik Pasien: Pembayaran

No	ID	Nama Pembayaran	Total
1	PG01	Pembayaran Cash Poliklinik Gigi (UMUM)	45.000,00

No Item: 1
ID: PG01
Total: 45.000,00

Total Transaksi: 45.000,00

Tambah, Ubah, Hapus, Batal, OK, Simpan, Tampil, Cetak, Desain, Keluar

Gambar 5.12 Form Pembayaran Tindakan Pasien

Tekan Simpan untuk memasukkan data Transaksi ke dalam Database. Untuk mencetak kwitansi transaksi tekan tombol Tampil untuk menampilkan Kwitansi ke layar komputer, Cetak untuk mencetak langsung ke Printer, dan Desain untuk mendesain kwitansi.

RSI Darus Syifa Benowo
Jl. Raya Benowo, Surabaya
Telpn : (031) 1234435

Nama Pasien : HENDRO GOENAWAN

Terbilang : Empat Puluh Lima Ribu Rupiah

Untuk Pembayaran : Pembersihan karang gigi tiap rahang

Rp. 45.000,00

10 Januari 2005

NANA

Tanggal Cetak : 10-01-2005
Jam. Cetak : 10:56:16

UMUM
No. RM : 069636

Gambar 5.13 Kwitansi Pembayaran Transaksi

5.2.3.4 Uji Coba Pelayanan Rawat Inap

Diasumsikan Pasien Baru Shanti Puruhita melakukan pendaftaran Rawat Inap untuk melakukan persalinan.

Untuk melakukan pendaftaran Rawat Inap kita masuk ke menu Rawat Inap -- Registrasi – Registrasi Rawat Inap. Kemudian akan tampil form pendaftaran rawat inap seperti dibawah ini.

Menu - [Registrasi - Rawat Inap]

File Edit View

Pendaftaran Pasien Rawat Inap

No. RM	Nama	Alamat	S. U	Kelurahan	Kecamatan	Masuk	Ruang	Kamar	Tipe Pasien
012379	ABD HARI	PERUM KEHAKIMAN E/20	L	53		05/01/2005	RAWAT	Kamar Kelas B.3	AKSES
012379	DANI	ASPIK P. B.	P	35		05/01/2005	RAWAT	Kamar Kelas B.15	AKSES

Baru

✓ Edit

✗ Hapus

✗ Cetak

✗ Lampir

✗ Deteksi

✗ Setup

✗ Input

✗ Keluar

10/01/2005 11:00 THUNDERA SUPERVISOR Nana PostgreSQL 3 Microsoft Of... SQL Script Delphi 6 Jacc Paint Shop ... Rsi_Jenawa 11:00 AM

Gambar 5.14 Form Pendaftaran Pasien Rawat Inap

Untuk melakukan pendaftaran pasien baru klik tombol Baru untuk menampilkan input pasien rawat baru, seperti dibawah ini :

Input Data Pasien

RM	AUTO_NUMBER						
Nama	SHAHTI PURUHITA	Jkl	P	Status	Menikah		
Tmp. Lhr	MALANG	Tgl. Lhr	05/12/1979	Umur	25	Gol Darah	AB
Alamat	JL. BARENG KARTINI IMB						
RT / RW	4	6	Desa/Kelurahan	BARENG	Kecamatan	KLOJEN	
Kota	MALANG	Kode Pos	34223	Telepon	(0341) 342445		
Agama	Islam	Pendidikan	S1	Pekerjaan	SIYASTA		
Catatan							
Suami	HAHAHNG HARIADI	Telepon	(0341) 342445				
Alamat	JL. BARENG KARTINI IMB						
Kontak	HAHAHNG HARIADI	Telepon					
Alamat							

Batal Simpan

Gambar 5.15 Form Input Pasien Rawat Inap

Setelah user melakukan pengisian data pasien, klik simpan untuk menyimpan data pasien baru, kemudian akan tampil Form Input Daftar Rawat Inap untuk memilih kamar, dokter yang merawat, ruangan perawatan, tipe pasien, uang muka perawatan, dan lain-lain.

Data Pasien Rawat Inap

Tanggal/Jam Pendaftaran 10/01/2005 11:15:01 AM

RM	AUTO_NUMBER				
Nama	SHAHTI PURUHITA	P	Lahir	05/12/1979	25
Alamat	JL. BARENG KARTINI IMB				
RT / RW	4	6	Desa / Kecamatan	BARENG	KLOJEN
Kota	MALANG	Gol Darah	AB		
Dept.ID	RAWAT	Kamar	Kamar Kelas I		Hapus
Kategori	UMUM	Uang Muka	Rp 1,000,000.00		
Dokter	DKT008 DAIHU MARYOTO TEGUH, DR. SP.GG				
Cara Mek.	TP2RI	Oleh			
DX	PERSALINAN		Kondisi Mek.	Lemas	

Batal Input Edit Data Pasien

Gambar 5.16 Form Input Pasien Rawat Inap

Setelah memasukkan data pasien rawat inap, kemudian klik input untuk memasukkan ke dalam daftar pasien rawat inap seperti dibawah

Main - [Registrasi - Registrasi Rawat Inap]

File Edit View

ini :

Pendaftaran Pasien Rawat Inap									
No. RM	Nama	Alamat	S	U	Kelurahan	Kecamatan	Masuk	Ruang	Tipe Pasien
012379	ABD HARI	PERU 4 KEHAKIMAN E/20	L	53			05/01/2005	RAWAT	Kamar Kelas II.3 ASKES
027535	DINA	ASPOL KP III	P	26			05/01/2005	RAWAT	Kamar Kelas III.18 UMUM
069637	SHANTI PURUHITA	JL BARENG PARTER III	P	35 BARENG	LOJEN		10/01/2005	RAWAT	Kamar Kelas I UMUM

Gambar 5.17 Form Pasien Rawat Inap Updated

Untuk mencetak kwitansi Uang Muka Rawat Inap, klik tombol Kuitansi di pojok kanan bawah, yang akan menghasilkan kwitansi seperti dibawah ini :

RSI Darus Syifa Benowo
Jl. Raya Benowo, Surabaya
Telpon : (031) 1234435

Nama Pasien

: SHANTI PURUHITA (069637)

Terbilang

: Satu Juta Rupiah

Untuk Pembayaran

: Pembayaran Uang Muka

Rp 1.000.000.00

10-01-2005

Jam Datang : 11:15:01

NANA

Jam Cetak :

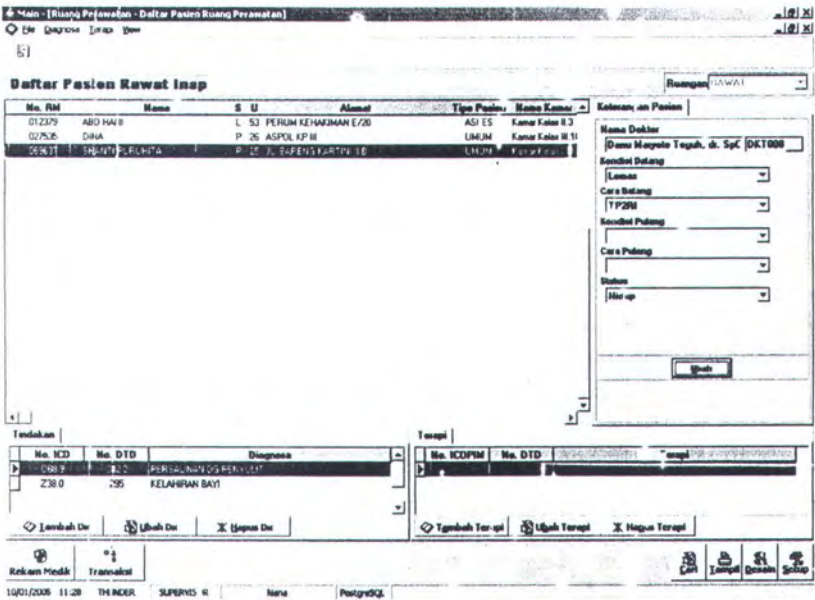
UMUM
No. RM : 069637

Gambar 5.18 Kwitansi Uang Muka Rawat Inap

Pasien menuju ke kamar perawatan untuk melakukan opname sampai sembuh, atau diijinkan pulang oleh rumah sakit.

Didalam kamar perawatan, setiap harinya dokter atau perawat akan melakukan pemeriksaan serta melakukan tindakan apabila diperlukan terhadap pasien. Perawat akan mencatatnya ke dalam komputer setiap selesai pemeriksaan atau tindakan. Form Daftar Pasien

Rawat Inap untuk memasukkan rekam medik dan tindakan adalah
sebagaimana berikut :



Gambar 5.19 Daftar Pasien Ruang Perawatan

Dalam Form ini terdapat beberapa fungsi antara lain :

1. Memasukkan Keterangan Pasien
Klik tombol Ubah untuk melakukan edit dokter yang merawat, kondisi datang, cara data, kondisi pulang serta cara pulang, serta status pasien ketika keluar.
2. Memasukkan Diagnosa Pasien
Klik tombol Tambah Dx untuk menambah Diagnosa Pasien.
3. Memasukkan Terapi Pasien
Klik tombol Tambah Terapi untuk menambah Terapi Pasien
4. Memasukkan Rekam Medik
Klik tombol Rekam Medik untuk menambah data Rekam Medik Pasien. Akan muncul Form Summary Rekam Medik ketika pasien

dirawat. Dalam Form ini user bisa menambah data rekam medik, mengubah data, dan menghapus data rekam medik.

Tanggal RM	Jam RM	Nama Dokter
10/01/2005	11:39:12 AM	Yanuar Syam, dr. SpBU
10/01/2005	5:40:02 PM	Yanuar Syam, dr. SpBU

Tambah Ubah Hapus

Gambar 5.20 Form Summary Rekam Medik

Untuk menambah dan mengubah data rekam medik mempunyai proses yang sama dengan rawat jalan.

5. Memasukkan Transaksi Tindakan

Klik tombol Transaksi untuk menambah data Transaksi Tindakan. Kemudian akan muncul Form Summary Transaksi Pasien. Dalam Form ini user bisa menambah data, mengubah data, dan menghapus data transaksi.

Tanggal Tx	Jam Tx	Ruangan	Total Transaksi
10/01/2005	11:50:34 AM	Ruang Bersalin	2,150,000.00
11/01/2005	12:00:10 PM	Laboratorium Rawat Inap	50,000.00
10/01/2005	12:00:41 PM	Ruang Perawatan	349,500.00

Tambah Ubah Hapus

Total Transaksi 2,529,500.00

Gambar 5.21 Form Summary Transaksi Tindakan

Klik tombol Tambah untuk menambah data Transaksi Tindakan Pasien. Pada Form Transaksi Tindakan Pasien, user juga dapat mengisi tindakan-tindakan dokter atau perawat yang melakukan tindakan.

TINDAKAN MEDIK PASIEN

Pada

Nama Pasien: SHANTI PURUHITA ID: 069637 Tipe Pasien: UMUM

Tanggal: 10/01/2005 11:50:34 AM Departemen: VK

Tindakan Medik Pasien

No	ID	Nama	Jml	Harga
1	TM019	Persalinan Spontaneus Vacuum Kelas I	1	1.900.000

No Item: 1

ID: M0119 Persalinan Spontaneus Vacuum Kelas I

Jumlah: 1 Harga: 1.900.000.00

Total: 1.900.000.00

Pelaksana Tindakan

ID	Nama Staff	Tipe Staff	Tipe Ts
DKT008	Danu Meryoto Teguh, dr. SpOG	Dokter	Operator
DKT003	Bambang Suhadi, dr.	Dokter	Anestesi
STF090	Aan Nuldiyanto, AMK	Perawat	Asisten

Total Transaksi: 1.900.000.00

Tambah Ubah Hapus Batal OK Simpan Tampil Cek Desain Keluar

Gambar 5.22 Form Transaksi Tindakan Pasien Rawat Inap

Tekan Simpan untuk memasukkan data Transaksi ke dalam Database.

6. Melihat Data Pemakaian Kamar Pasien

Klik tombol View – Pemakaian kamar untuk melihat data pemakaian kamar pasien.

PEMAKAIAN KAMAR

Pemakaian Kamar Pasien

Nama Kamar	Kelas	Tgl Masuk	Jam Masuk	Tgl Keluar	Jam Keluar	Lama	Deskripsi	Biaya Kamar
Kamar Kelas I	Kelas I	10/01/2005	11:15:01 AM					Y

Gambar 5.23 Form Pemakaian Kamar

Untuk memasukkan data obat-obatan yang dipakai selama perawatan, masuk Kembali ke Menu Utama Klik Apotik – Apotik Rawat Inap – Penjualan Obat Rawat Inap. Kemudian akan muncul form pembelian obat rawat inap. Tekan Menu File - Add Pembelian Obat, kemudian tekan tombol tambah untuk mengisi data obat yang ditransaksikan tapi terlebih dahulu memilih data pasien rawat inap yang akan membeli obat.

Data dari Transaksi obat ini dapat dilihat pada form Daftar Pasien Rawat Inap di ruangan tempat perawatan pasien. Gambar dari form Transaksi Obat Rawat Inap adalah sebagaimana berikut :

Form Transaksi Obat Pasien Rawat Inap

Nama Pasien: SHANTI PURUHITA ID: 069637 Tipe Pasien: UMUM
 Nama Dokter: Danu Maryoto Teguh, dr. SpOG DKT008
 Tanggal: 10/01/2005 12:05:18 PM Jenis Pembelian: BEBAS

No	ID	Nama	Jml	Harga	Disk.	R	Total
1	ATS02	ATS visl 20.000	1	272.250.00	0	0	272.250.00
2	AM006	Amoxicap 500 mg	30	3.094.00	0	0	92.820.00
3	ACY03	Acyclovir salep 5 %	1	4.079.00	0	0	4.079.00
4	AL102	Alineman F Odolletab 5 mg	15	680.00	0	0	10.200.00

No Rem: 4
 ID: AL102
 Alineman F Odolletab 5 mg
 Jumlah: 15
 Harga: 680.00
 Diskon: 0 % 0.00
 Total Sbm. Diskon: 10.200.00
 Total Diskon: 0.00
 R: 0.00
 Total: 11.200.00

Total Transaksi: 379.349.00

Tambah Ubah Hapus Simpan Cetak Desain Setup Keluar

10/01/2005 12:06 THUNDER SUPERVISOR: Nana PostgreSQL
 Start Tuner Apak... Tuner Contr... 3 Microsof... SQL Script Delphi 6 Ral_benowo 12:06 PM

Gambar 5.24 Form Transaksi Obat Pasien Rawat Inap

Setelah selesai melakukan pengisian data obat, tekan Simpan untuk menyimpan data obat ke database.

Untuk melihat data transaksi obat yang sudah dilakukan pasien rawat inap, dapat dilihat di Form Daftar Pasien Rawat Inap, menu View -- Transaksi Obat, seperti pada cuplikan form dibawah ini

Daftar Transaksi Obat						
Daftar Transaksi Obat Pasien						
Tanggal Tx	Jam Tx	Nama Pasien	Alamat Pasien	Dokter	Jenis Pembelian	Total
10/01/2005	12:04:08 PM	SHANTI PURUHITA	JL. BARENG KARTINI IIB	Danu Mulyoto Teguh	BEBAS	645.433.00

Gambar 5.25 Form Transaksi Obat Pasien Rawat Inap

Setelah pasien sembuh, maka pasien menuju kasir rawat inap untuk membayar seluruh data perawatan rawat inap. Untuk menampilkan data serta melakukan transaksi pembayaran rawat inap pasien dengan menggunakan Form Pembayaran Transaksi Rawat Inap, klik Rawat Inap -- Pembayaran -- Pembayaran Pulang Rawat Inap, kemudian akan muncul Form seperti dibawah ini :

Main - [Pembayaran - Pembayaran Pulang Rawat Inap]

File View

Nama Pasien: SHANTI PURUHITA 06963 Tipe Pasien: UMUM

Alamat: JL. BARENG KARTINI IIB

Nama Dokter: Danu Mulyoto Teguh, dr. Sp.06 DKT.008

Tanggal Masuk: 10/01/2005 12:34:44 PM Tanggal Pulang: 10/01/2005 12:34:44 PM

Kamar: Kamar Kelas I Kelas: KELAS I

Transaksi Pasien Rawat Inap: Pembayaran

No.	Deskripsi	Total	Revisi Layout
1	Laboratorium	30,000.00 A17	
2	Visite Dokter	130,000.00 A04	
3	Tindakan Medis Non Operatif	72,000.00 / 07	
4	Persalinan	2,150,000.00 A08	
5	Biaya Administrasi	10,000.00 A26	
6	Ambulance	17,500.00 / 38	
7	Darah	120,000.00 A29	
8	Kamar	75,000.00 A01	
9	Obat	379,349.00 A01	

Total Transaksi: 2,963,849.00 Uang Muka: 1,000,000.00 Sisa Pembayaran: 1,963,849.00

Tambah Hapus OK Simpan Tampil Cetak Desain Setup Ekspor

10/01/2005 12:36 THUNDER SUPERVISOR Nana PostgreSQL

Start 2 pc tv 3 Microsoft Of ... Control Panel Delphi 6 Jasc Paint Shop ... Rsl_benowo 12:36 PM

Gambar 5.26 Form Pembayaran Pulang Rawat Inap

Form ini akan menampilkan seluruh transaksi pasien rawat inap selama pasien dirawat, dari biaya tindakan, kamar, obat dan alat kesehatan.

Untuk pembayaran pasien rawat inap prosesnya sama dengan pembayaran tindakan pasien rawat jalan.

Ketika disimpan pasien otomatis akan dihapus dari form pendaftaran pasien rawat inap, kemudian dipindahkan ke form history rawat inap. Yang digunakan untuk melihat kembali aktifitas-aktifitas rawat inap pasien.

5.2.3.5 Uji Coba Pelayanan Apotek Umum

Diasumsikan pasien luar Firman dengan alamat Jl. Menur Pumpungan melakukan pembelian obat di apotek, dengan dokter yang memberikan resep adalah Hadi S. dr.

Untuk memasukkan data transaksi obat umum, klik pada Menu Utama Apotik – Apotik Rawat Jalan – Penjualan Obat Rawat Jalan, kemudian akan muncul Form seperti dalam gambar 5.27.

Untuk memasukkan data pasien luar, user langsung mengetikkan nama, alamat, dan dokter secara langsung. Untuk pasien yang terdaftar dengan memasukkan nomor rekam medi'k pasien.

Untuk pembayaran apotek prosesnya sama dengan pembayaran tindakan pasien rawat jalan serta pembayaran rawat inap.

Main - (Apotek Rawat Jalan - Penjualan Obat Rawat Jalan)

File View

Nama Pasien: FIRMAN Tipe Pasien: UMUM
 Alamat: JL. MENUR PUMUNGAN
 Nama Dokter: HADI S. dr.
 Tanggal: 10/01/2005 12:47:26 PM Jenis Pembayaran: BEBAS

Transaksi Obat Pasien | Pembayaran

No	ID	Nama	Jml	Harga	Disk.	R	Total
1	AM006	Amoxon cap 500 mg	25	3.094.00	0	0	77.350.00
2	ACT02	Acidol lab	15	1.230.00	0	0	18.450.00
3	AM004	Amoxon cap 125 mg/5 ml x 60	1	3.537.00	0	0	3.537.00

Total Transaksi: 99.737.00

Tambah Ubat Hapus Batas Simpan Cetak Desain Setup Ekspor

10/01/2006 12:53 THUNDER SUPERVISOR Nana PostgreSQL

Start 2 pc-tv 3 Microsoft Of... File Delphi 6 Jasc Paint Shop ... Rsl_benowo 12:53 PM

Gambar 5.27 Form Penjualan Obat Umum

5.2.3.6 Uji Coba Pelayanan Pembayaran Hutang

Diasumsikan pasien Shanti Puruhita dengan nomor rekam medik 069637 akan membayar hutang pembayaran rawat inap.

Untuk memasukkan data pembayaran hutang pasien, klik pada Menu Utama System – Hutang – Pembayaran Hutang, kemudian akan muncul Form seperti gambar 5.28.

Untuk memasukkan data pasien yang akan membayar, user memasukkan nomor rekam medik pasien. Kemudian memasukkan data hutang yang akan dibayar. Tekan Simpan untuk memasukkan ke dalam database.

Main : [Hutang - Pembayaran Hutang]

File View

1

Nama Pasien: SHANTI PURUHITA ID: 069637

Alamat: JL. BARENG KARTINI III

Tanggal: 10/01/2005 12:51:53 PM

Pembayaran Hutang

No	ID	Keterangan	Total	Bayar
1	HTNG_ID/2005/7	Bayar Lunas	903.849	903.849

No Rem

ID: HTNG_ID/2005/7

Ket: Bayar Lunas

Bayar: 903.849.00

Total Pembayaran: 903.849.00

Tambah Hapus Simpan Tampil Cetak Desain Setup Keluar

10/01/2005 12:59 THUNDER SUPERVISOR Nana PostgreSQL

Start 2 pc-tv Microsoft Of... Fikri Delphi 6 Jasc Paint Shop ... Rsl_benowo 1:00 PM

Gambar 5.28 Form Pembayaran Hutang

5.2.3.7 Uji Coba Pencetakan Laporan Rumah Sakit

Adapun uji coba pencetakan laporan rumah sakit, hasilnya dapat dilihat pada lampiran di bagian belakang buku.

5.2.4 Hasil Uji Coba

Dari uji coba terhadap aplikasi Sistem Informasi Pasien dan Rekam Medik yang telah dibuat, didapat hasil bahwa aplikasi dapat menjalankan seluruh fungsi yang ada padanya dengan baik dan telah memenuhi kebutuhan untuk mencatat segala aktifitas pasien ketika di rumah sakit. Mungkin kekurangannya belum dapat mencatat segala aktifitas rumah sakit.

BAB VI

PENUTUP

Pada bab ini akan dijelaskan kesimpulan yang bisa diambil dari Tugas Akhir ini serta kemungkinan pengembangan yang bisa dilakukan terhadap aplikasi Sistem Informasi Pasien dan Rekam Medik yang telah dibuat.

6.1 Kesimpulan

Dari hasil seluruh analisa, desain, implementasi, serta uji coba aplikasi Sistem Informasi Pasien dan Rekam Medik ini dapat diambil kesimpulan sebagai berikut :

1. Sistem Informasi Pasien dan Rekam Medik yang dihasilkan dapat memenuhi kebutuhan Rumah Sakit Islam Darus Syifa' akan segala informasi-informasi yang berkaitan dengan pasien.
2. Sistem Informasi Pasien dan Rekam Medik yang dihasilkan dapat meningkatkan pelayanan pasien di setiap bagian rumah sakit, misal di kamar terima terjadi pengurangan antrian pendaftaran pasien karena adanya fasilitas pencarian data pasien secara cepat.
3. Sistem yang dihasilkan memudahkan Perawat atau Dokter dalam melakukan perawatan pasien, dimana Perawat atau Dokter dapat dengan mudah melihat informasi daftar pasien yang dirawat serta melihat catatan rekam medik pasien terdahulu.

- TObatRI_Dtl

Entitas ini berisi data-data transaksi obat pasien rawat jalan dan transaksi obat oleh pasien luar

```
CREATE TABLE TObatRI_Dtl
(
  TORI_ID          VARCHAR(32) REFERENCES TObatRI
                  ON UPDATE CASCADE ON DELETE CASCADE,
  Noltem           INT DEFAULT 0,
  ObatID           VARCHAR(32) REFERENCES Cbat
                  ON UPDATE CASCADE ON DELETE SET NULL,
  Jumlah           FLOAT8 DEFAULT 0,
  HDJ              FLOAT8 DEFAULT 0,
  FHDJ             FLOAT8 DEFAULT 0,
  Harga            FLOAT8 DEFAULT 0,
  HDD              FLOAT8 DEFAULT 0,
  FHDD             FLOAT8 DEFAULT 0,
  Diskon           FLOAT8 DEFAULT 0,
  NilaiR           FLOAT8 DEFAULT 0,
  Total            FLOAT8 DEFAULT 0
);
GRANT ALL PRIVILEGES ON TObatRI_Dtl TO PUBLIC;
```

- TOPasien

Entitas ini digunakan untuk mencatat pasien rumah sakit yang membeli obat di apotik rawat jalan rumah sakit.

```
CREATE TABLE TOPasien
(
  TO_ID            VARCHAR(32) REFERENCES TObatUmum
                  ON UPDATE CASCADE ON DELETE CASCADE,
  PasienID         VARCHAR(250) REFERENCES Pasien
                  ON UPDATE CASCADE ON DELETE CASCADE,
  TipePasien       VARCHAR(250) REFERENCES TipePasien
                  ON UPDATE CASCADE ON DELETE SET NULL,
  PRIMARY KEY(TO_ID, PasienID)
);
GRANT ALL PRIVILEGES ON TOPasien TO PUBLIC;
```

- TObatDokter

Entitas ini digunakan untuk mencatat dokter rumah sakit yang memberikan resep kepada pasien.

```
CREATE TABLE TObatDokter
(
  TO_ID            VARCHAR(32) REFERENCES TObatUmum
                  ON UPDATE CASCADE ON DELETE CASCADE,
  DokterID         VARCHAR(32) REFERENCES Dokter
                  ON UPDATE CASCADE ON DELETE SET NULL,
  PRIMARY KEY (TO_ID, DokterID)
);
GRANT ALL PRIVILEGES ON TODokter TO PUBLIC;
```

- SetHargaJual

Entitas ini digunakan untuk menentukan harga jual obat sesuai dengan jenis pembelian dan tipe pasien yang dilayani oleh apotek.

```
CREATE TABLE SetHargaJual
(
  RecID            SERIAL NOT NULL,
  JenisPbl_ID      VARCHAR(32) REFERENCES JenisPembelian
                  ON UPDATE CASCADE ON DELETE CASCADE,
  TipePasienID     VARCHAR(32) REFERENCES TipePasien
                  ON UPDATE CASCADE ON DELETE CASCADE,
  HDJ              INT DEFAULT 0,
  HDJFaktor        FLOAT8 DEFAULT 1,
  HDD              INT DEFAULT 0,
  Diskon           FLOAT8 DEFAULT 0,
  PRIMARY KEY(RecID)
);
```



```
GRANT ALL PRIVILEGES ON SetHargaJual TO PUBLIC;
GRANT ALL PRIVILEGES ON SetHargaJual_ReclD_Seq TO PUBLIC;
```

- **Karcis**

Entitas ini berisi data-data karcis masuk rumah sakit untuk rawat jalan

```
CREATE TABLE Karcis
(
  Karcis_ID          VARCHAR(32) PRIMARY KEY,
  Karcis_Nama        VARCHAR(250),
  Dept_ID            VARCHAR(32) REFERENCES Departemen
                    ON UPDATE CASCADE ON DELETE CASCADE,
  TK_ID              VARCHAR(32) REFERENCES TipeKunjungan
                    ON UPDATE CASCADE ON DELETE SET NULL,
  TP_ID              VARCHAR(32) REFERENCES TipePasien
                    ON UPDATE CASCADE ON DELETE CASCADE,
  Harga              FLOAT8 DEFAULT 0
);
GRANT ALL PRIVILEGES ON Karcis TO PUBLIC;
```

- **Kelas**

Entitas ini berisi kelas-kelas perawatan yang ada di rumah sakit

```
CREATE TABLE Kelas
(
  Kelas_ID           VARCHAR(32) PRIMARY KEY,
  Kelas_Nama          VARCHAR(250),
  Kelas_Level         INT
);
GRANT ALL PRIVILEGES ON Kelas TO PUBLIC;
```

- **Kamar**

Entitas ini berisi data-data kamar rawat inap yang terdapat di rumah sakit

```
CREATE TABLE Kamar
(
  Kamar_ID           VARCHAR(32) PRIMARY KEY,
  Kamar_Nama          VARCHAR(250),
  DepartemenID        VARCHAR(32) REFERENCES Departemen
                    ON UPDATE CASCADE ON DELETE SET NULL,
  IsActive            VARCHAR(1) DEFAULT 'N',
  Kelas_ID            VARCHAR(32) REFERENCES Kelas
                    ON UPDATE CASCADE ON DELETE SET NULL,
  TipeID              VARCHAR(32),
  IsCheckIn            INT,
  JamMasuk             TIME,
  IsCheckOut           INT,
  JamKeluar            TIME,
  Hour                FLOAT4 DEFAULT 0,
  Harga               FLOAT8 DEFAULT 0
);
GRANT ALL PRIVILEGES ON Kamar TO PUBLIC;
```

- **TransaksiTx_Dtl**

Entitas ini berisi data-data transaksi tindakan detail.

```
CREATE TABLE TransaksiTx_Dtl
(
  TT_Dtl             VARCHAR(32) NOT NULL PRIMARY KEY,
  TT_ID              VARCHAR(32) REFERENCES TransaksiTx
                    ON UPDATE CASCADE ON DELETE CASCADE,
  Noltem             INT DEFAULT 0,
  TindakanID         VARCHAR(32) REFERENCES Tindakan
                    ON UPDATE CASCADE ON DELETE SET NULL,
  Jumlah             FLOAT8 DEFAULT 0,
  Harga              FLOAT8 DEFAULT 0,
  Total              FLOAT8 DEFAULT 0
);
GRANT ALL PRIVILEGES ON TransaksiTx_Dtl TO PUBLIC;
```

- **TransaksiTxRI**

Entitas ini berisi data-data transaksi tindakan yang ada di rawat inap.

```
CREATE TABLE TransaksiTxRI
```

```
(
  TT_ID      VARCHAR(32) REFERENCES TransaksiTx
            ON UPDATE CASCADE ON DELETE CASCADE,
  PRI_ID     VARCHAR(32) REFERENCES PasienRI
            ON UPDATE CASCADE ON DELETE CASCADE,
  Dept_ID    VARCHAR(32) REFERENCES Departemen
            ON UPDATE CASCADE ON DELETE SET NULL,
  PRIMARY KEY(TT_ID,PRI_ID)
);
GRANT ALL PRIVILEGES ON TransaksiTxRI TO PUBLIC;
```

- **TransaksiTindakanRJ**

Entitas ini berisi data-data transaksi tindakan yang ada di rawat jalan.

```
CREATE TABLE TransaksiTxRJ
(
  TT_ID      VARCHAR(32) REFERENCES TransaksiTx
            ON UPDATE CASCADE ON DELETE CASCADE,
  DPRJ_ID    VARCHAR(32) REFERENCES PasienRJ
            ON UPDATE CASCADE ON DELETE CASCADE,
  PRIMARY KEY(TT_ID,DPRJ_ID)
);
GRANT ALL PRIVILEGES ON TransaksiTindakanRJ TO PUBLIC;
```

- **TxDokter**

Entitas ini bertujuan untuk mengetahui dokter mana saja yang memberikan tindakan pada pasien.

```
CREATE TABLE TxDokter
(
  TT_Dtl     VARCHAR(32) REFERENCES TransaksiTx_Dtl
            ON UPDATE CASCADE ON DELETE CASCADE,
  DokterID   VARCHAR(32) REFERENCES Dokter
            ON UPDATE CASCADE ON DELETE CASCADE,
  TxTipe     INT,
  PRIMARY KEY (TT_Dtl, DokterID)
);
GRANT ALL PRIVILEGES ON TxDokter TO PUBLIC;
```

- **TxStaff**

Entitas ini bertujuan untuk mengetahui dokter mana saja yang memberikan tindakan pada pasien.

```
CREATE TABLE TxStaff
(
  TT_Dtl     VARCHAR(32) REFERENCES TransaksiTx_Dtl
            ON UPDATE CASCADE ON DELETE CASCADE,
  StaffID    VARCHAR(32) REFERENCES Staff
            ON UPDATE CASCADE ON DELETE CASCADE,
  TxTipe     INT,
  PRIMARY KEY (TT_Dtl, StaffID)
);
GRANT ALL PRIVILEGES ON TxStaff TO PUBLIC;
```

- **RekamMedik**

Entitas ini berisi data-data rekam medik pasien.

```
CREATE TABLE RekamMedik
(
  RM_ID      VARCHAR(32) NOT NULL PRIMARY KEY,
  Dokter_ID  VARCHAR(32) REFERENCES Dokter
            ON UPDATE CASCADE ON DELETE SET NULL,
  Tanggal    DATE DEFAULT CURRENT_DATE,
  Jam        TIME DEFAULT CURRENT_TIME
);
GRANT ALL PRIVILEGES ON RekamMedik TO PUBLIC;
```

- **RekamMedik_Dtl**

Entitas ini berisi data-data detail dari rekam medik.

```
CREATE TABLE RekamMedik_Dtl
(
  RM_ID      VARCHAR(32) REFERENCES RekamMedik
```



```

RMTipe          VARCHAR(10),
RMTipeInt       INT,
TipeDeskripsi   SMALLINT,
TipeData        SMALLINT,
Deskripsi       VARCHAR(250),
NilaiNormal     VARCHAR(250),
Satuan          VARCHAR(250),
ResultString     VARCHAR(250),
ResultFloat     FLOAT4,
DataMinStd      FLOAT4,
DataMaxStd      FLOAT4
);
GRANT ALL PRIVILEGES ON RekamMedik_Dtl TO PUBLIC;;

```

- **RekamMedikRI**

Entitas ini berisi data-data RekamMedikPasien yang ada di rawat inap.

```

CREATE TABLE RekamMedikRI
(
  RM_ID          VARCHAR(32) REFERENCES RekamMedik
                ON UPDATE CASCADE ON DELETE CASCADE,
  PRI_ID         VARCHAR(32) REFERENCES PasienRI
                ON UPDATE CASCADE ON DELETE CASCADE,
  PRIMARY KEY(RM_ID,PRI_ID)
);
GRANT ALL PRIVILEGES ON RekamMedikRI TO PUBLIC;

```

- **RekamMedikRJ**

Entitas ini berisi data-data Rekam Medik Pasien yang ada di rawat jalan.

```

CREATE TABLE RekamMedikPasienRJ
(
  RM_ID          VARCHAR(32) REFERENCES RekamMedikPasien
                ON UPDATE CASCADE ON DELETE CASCADE,
  DPRJ_ID        VARCHAR(32) REFERENCES DaftarPasienRawatJalan
                ON UPDATE CASCADE ON DELETE CASCADE,
  PRIMARY KEY(RM_ID,DPRJ_ID)
);
GRANT ALL PRIVILEGES ON RekamMedikPasienRJ TO PUBLIC;

```

- **DiagnosaRJ**

Entitas ini berisi data-data diagnosa pasien rawat jalan

```

CREATE TABLE DiagnosaRJ
(
  DxRJ_ID        VARCHAR(32) NOT NULL PRIMARY KEY,
  PRJ_ID         VARCHAR(32) REFERENCES PasienRJ
                ON UPDATE CASCADE ON DELETE CASCADE,
  JenisKasus     VARCHAR(32),
  DxTipe         VARCHAR(32),
  ICD            VARCHAR(64) REFERENCES ICD
                ON UPDATE CASCADE ON DELETE SET NULL
);
GRANT ALL PRIVILEGES ON DiagnosaRJ TO PUBLIC;;

```

- **TerapiRJ**

Entitas ini berisi data-data terapi pasien rawat jalan

```

CREATE TABLE TerapiRJ
(
  TRJ_ID         VARCHAR(32) NOT NULL PRIMARY KEY,
  PRJ_ID         VARCHAR(32) REFERENCES PasienRJ
                ON UPDATE CASCADE ON DELETE CASCADE,
  ICOPIM         VARCHAR(64) REFERENCES ICD
                ON UPDATE CASCADE ON DELETE SET NULL
);
GRANT ALL PRIVILEGES ON TerapiRJ TO PUBLIC;

```

- **PemakaianKamar**

Entitas ini berisi data-data pemakaian kamar rawat inap oleh pasien, dimana setiap kegiatan pasien dalam menggunakan kamar baik masuk atau keluar dari kamar akan dicatat di tabel ini

```
CREATE TABLE PemakaianKamar
(
  PMKK_ID          VARCHAR(32) NOT NULL PRIMARY KEY,
  Kamar_ID         VARCHAR(32) REFERENCES Kamar
                  ON UPDATE CASCADE ON DELETE SET NULL,
  PRI_ID           VARCHAR(32) REFERENCES PasienRI
                  ON UPDATE CASCADE ON DELETE CASCADE,
  TanggalMasuk     DATE,
  JamMasuk         TIME,
  TanggalKeluar     DATE,
  JamKeluar        TIME,
  LamaPerawatan    INT,
  IDKeluar          INT,
  DeskripsiKeluar   VARCHAR(32),
  DepartemenIDBaru VARCHAR(32),
  NamaDepartemenBaru VARCHAR(250),
  IDKamarBaru      VARCHAR(32),
  NamaKamarBaru    VARCHAR(250),
  IsAktif           VARCHAR(1) DEFAULT 'Y'
);
```

```
GRANT ALL PRIVILEGES ON PemakaianKamar TO PUBLIC;
```

- TerapiRI

Entitas ini berisi data-data terapi pasien rawat inap

```
CREATE TABLE TerapiRI
(
  TRI_ID          VARCHAR(32) NOT NULL PRIMARY KEY,
  PRI_ID          VARCHAR(32) REFERENCES PasienRI
                  ON UPDATE CASCADE ON DELETE CASCADE,
  ICOPIM          VARCHAR(64) REFERENCES ICM
                  ON UPDATE CASCADE ON DELETE SET NULL
);
```

```
GRANT ALL PRIVILEGES ON TerapiRI TO PUBLIC;
```

- DiagnosaRI

Entitas ini berisi data-data diagnosa pasien rawat inap

```
CREATE TABLE DiagnosaRI
(
  DxRI_ID        VARCHAR(32) NOT NULL PRIMARY KEY,
  PRI_ID         VARCHAR(32) REFERENCES PasienRI
                  ON UPDATE CASCADE ON DELETE CASCADE,
  JenisKasus     VARCHAR(32),
  DxTipe         VARCHAR(32),
  ICD            VARCHAR(64) REFERENCES ICD
                  ON UPDATE CASCADE ON DELETE SET NULL
);
```

```
GRANT ALL PRIVILEGES ON DiagnosaRI TO PUBLIC;
```

- DeptStaff

Entitas ini berisi informasi staff bekerja di departemen apa di rumah sakit.

```
CREATE TABLE DeptStaff
(
  StaffID        VARCHAR(32) REFERENCES Staff
                  ON UPDATE CASCADE ON DELETE CASCADE,
  DepartemenID   VARCHAR(32) REFERENCES Departemen
                  ON UPDATE CASCADE ON DELETE CASCADE,
  PRIMARY KEY (StaffID, DepartemenID)
);
```

```
GRANT ALL PRIVILEGES ON DeptStaff TO PUBLIC;
```

- DeptDokter

Entitas ini berisi informasi dokter bekerja di departemen apa di rumah sakit.

```
CREATE TABLE DeptDokter
```


- ```
(
 DokterID VARCHAR(32) REFERENCES Dokter
 ON UPDATE CASCADE ON DELETE CASCADE,
 DepartemenID VARCHAR(32) REFERENCES Departemen
 ON UPDATE CASCADE ON DELETE CASCADE,
 PRIMARY KEY (DokterID, DepartemenID)
);
GRANT ALL PRIVILEGES ON DeptDokter TO PUBLIC;
```
- **DeptTindakan**  
Entitas ini bertujuan untuk mengetahui dan mengelompokkan tindakan-tindakan medis sesuai dengan poliklinik atau ruang perawatan di rumah sakit.  
CREATE TABLE DeptTindakan  
(  
 TindakanID VARCHAR(32) REFERENCES Tindakan  
 ON UPDATE CASCADE ON DELETE CASCADE,  
 DepartemenID VARCHAR(32) REFERENCES Departemen  
 ON UPDATE CASCADE ON DELETE CASCADE,  
 PRIMARY KEY (TindakanID, DepartemenID)  
);  
GRANT ALL PRIVILEGES ON DeptTindakan TO PUBLIC;
  - **DeptICD**  
Entitas ini bertujuan untuk mengetahui dan mengelompokkan kode ICD sesuai dengan poliklinik atau ruang perawatan di rumah sakit.  
CREATE TABLE DeptICD  
(  
 ICD VARCHAR(64) REFERENCES ICD  
 ON UPDATE CASCADE ON DELETE CASCADE,  
 DepartemenID VARCHAR(32) REFERENCES Departemen  
 ON UPDATE CASCADE ON DELETE SET NULL,  
 PRIMARY KEY (ICD, DepartemenID)  
);  
GRANT ALL PRIVILEGES ON DeptICD TO PUBLIC;
  - **DeptICM**  
Entitas ini bertujuan untuk mengetahui dan mengelompokkan kode ICM sesuai dengan poliklinik atau ruang perawatan di rumah sakit.  
CREATE TABLE DeptICM  
(  
 ICOPIM VARCHAR(64) REFERENCES ICM  
 ON UPDATE CASCADE ON DELETE CASCADE,  
 DepartemenID VARCHAR(32) REFERENCES Departemen  
 ON UPDATE CASCADE ON DELETE SET NULL,  
 PRIMARY KEY (ICOPIM, DepartemenID)  
);  
GRANT ALL PRIVILEGES ON DeptICM TO PUBLIC;
  - **RincianTransaksiRI**  
Entitas ini berisi data-data rincian transaksi rawat inap pasien.  
CREATE TABLE RincianTransaksiRI  
(  
 RTRI\_ID VARCHAR(32) NOT NULL PRIMARY KEY,  
 PRI\_ID VARCHAR(32) REFERENCES PasienRI  
 ON UPDATE CASCADE ON DELETE CASCADE,  
 NoItem INT DEFAULT 0,  
 Deskripsi VARCHAR(250),  
 Total FLOAT8 DEFAULT 0  
);  
GRANT ALL PRIVILEGES ON RincianTransaksiRI TO PUBLIC;
  - **JenisPembayaran**  
Entitas ini berisi jenis-jenis pembayaran transaksi yang ada di rumah sakit  
CREATE TABLE JenisPembayaran

```
(
 JenisPbyr_iD VARCHAR(32) PRIMARY KEY,
 JenisPbyr_Nama VARCHAR(250)
);
GRANT ALL PRIVILEGES ON JenisPembayaran TO PUBLIC;
```

- Pembayaran

Entitas ini berisi macam-macam pembayaran transaksi yang bisa dipilih oleh pasien sesuai dengan jenis pembayaran, departemen dan tipe pasien di rumah sakit.

```
CREATE TABLE Pembayaran
(
 Pbyr_ID VARCHAR(32) NOT NULL PRIMARY KEY,
 Pbyr_Nama VARCHAR(250),
 JenisPbyr_ID VARCHAR(32) REFERENCES JenisPembayaran
 ON UPDATE CASCADE ON DELETE CASCADE,
 Dept_ID VARCHAR(32) REFERENCES Departemen
 ON UPDATE CASCADE ON DELETE CASCADE,
 TipePasien_ID VARCHAR(32) REFERENCES TipePasien
 ON UPDATE CASCADE ON DELETE CASCADE,
 Pbyr_Tipe INT
);
GRANT ALL PRIVILEGES ON Pembayaran TO PUBLIC;
```

- TransaksiPbyr

Entitas ini berisi data-data transaksi pembayaran yang dilakukan oleh pasien.

```
CREATE TABLE TransaksiPbyr
(
 TPbyr_ID VARCHAR(32) NOT NULL PRIMARY KEY,
 Tanggal DATE DEFAULT CURRENT_DATE,
 Jam TIME DEFAULT CURRENT_TIME
);
GRANT ALL PRIVILEGES ON TransaksiPbyr TO PUBLIC;
```

- TransaksiPbyr Dtl

Entitas ini berisi detail dari transaksi pembayaran.

```
CREATE TABLE TransaksiPbyr_Dtl
(
 TPbyr_ID VARCHAR(32) REFERENCES TransaksiPbyr
 ON UPDATE CASCADE ON DELETE CASCADE,
 PembayaranID VARCHAR(32) REFERENCES Pembayaran
 ON UPDATE CASCADE ON DELETE CASCADE,
 Noltem INT DEFAULT 0,
 Total FLOAT8 DEFAULT 0
);
GRANT ALL PRIVILEGES ON TransaksiPbyr_Dtl TO PUBLIC;
```

- TPbyrRJ

Entitas ini berisi daftar transaksi pembayaran pasien rawat jalan.

```
CREATE TABLE TPbyrRJ
(
 TPbyr_ID VARCHAR(32) REFERENCES Trar :aksiPbyr
 ON UPDATE CASCADE ON DELETE CASCADE,
 PRJ_ID VARCHAR(32) REFERENCES PasienRJ
 ON UPDATE CASCADE ON DELETE CASCADE,
 PRIMARY KEY(TPbyr_ID, PRJ_ID)
);
```

- TPbyrRI

Entitas ini berisi daftar transaksi pembayaran pasien rawat inap.

```
CREATE TABLE TPbyrRI
(
 TPbyr_ID VARCHAR(32) REFERENCES TransaksiPbyr
 ON UPDATE CASCADE ON DELETE CASCADE,
```



```

PRI_ID VARCHAR(32) REFERENCES PasienRI
 ON UPDATE CASCADE ON DELETE CASCADE,
PRIMARY KEY(TPbyr_ID, PRI_ID)
);
GRANT ALL PRIVILEGES ON TPbyrRI TO PUBLIC;

```

- TPbyrApt

Entitas ini berisi daftar transaksi pembayaran obat di apotik.

```

CREATE TABLE TPbyrApt
(
 TPbyr_ID VARCHAR(32) REFERENCES TransaksiPbyr
 ON UPDATE CASCADE ON DELETE CASCADE,
 TO_ID VARCHAR(32) REFERENCES TObatUmum
 ON UPDATE CASCADE ON DELETE CASCADE,
PRIMARY KEY(TPbyr_ID, TO_ID)
);
GRANT ALL PRIVILEGES ON TPbyrApt TO PUBLIC;

```

- Hutang

Entitas ini berisi daftar piutang rumah sakit.

```

CREATE TABLE Hutang
(
 Hutang_ID VARCHAR(32) NOT NULL PRIMARY KEY,
 TPbyr_ID VARCHAR(32) REFERENCES TransaksiPbyr
 ON UPDATE CASCADE ON DELETE CASCADE,
 Total FLOAT8 DEFAULT 0,
 Terbayar FLOAT8 DEFAULT 0,
 Tanggal DATE DEFAULT CURRENT_DATE,
 Jam TIME DEFAULT CURRENT_TIME
);
GRANT ALL PRIVILEGES ON Hutang TO PUBLIC;

```

- PbyrHutang

Entitas ini berisi data pembayaran piutang rumah sakit.

```

CREATE TABLE PbyrHutang
(
 PBH_ID VARCHAR(32) NOT NULL PRIMARY KEY,
 Tanggal DATE DEFAULT CURRENT_DATE,
 Jam TIME DEFAULT CURRENT_TIME
);
GRANT ALL PRIVILEGES ON PbyrHutang TO PUBLIC;

```

- PbyrHutang\_Dtl

Entitas ini berisi detail dari data pembayaran piutang rumah sakit.

```

CREATE TABLE PbyrHutang_Dtl
(
 PbyrHutang_ID VARCHAR(32) REFERENCES PbyrHutang
 ON UPDATE CASCADE ON DELETE CASCADE,
 HutangID VARCHAR(32) REFERENCES Hutang
 ON UPDATE CASCADE ON DELETE CASCADE,
 Noltem INT DEFAULT 0,
 Keterangan VARCHAR(250),
 Total FLOAT8 DEFAULT 0,
 Bayar FLOAT8 DEFAULT 0
);
GRANT ALL PRIVILEGES ON PbyrHutang_Dtl TO PUBLIC;

```



## Desain Forms

- Form Free Form Diagnosa

- Form Pembayaran Hutang

- Form Cetak Karcis



- Form Master ICD dan ICOPIM

Daftar ICD

| No.ICD | No.DTD | Sebab | Diagnosa |
|--------|--------|-------|----------|
|--------|--------|-------|----------|

Select

- Form Kartu Induk Utama Pasien

Kartu Induk Utama Pasien

Daftar Pasien

| No. HM | Nama Pasien | S U | Alamat |
|--------|-------------|-----|--------|
|--------|-------------|-----|--------|

Input Data Pasien

PMS

Nama

JkL

Tmp. Lhr

Tgl. Lhr. / /

Umur

Gol Darah

Status

Alamat

RT / RW

Kelurahan

Kecamatan

Kota

Kode Pos

Telepon

Agama

Pendidikan

Pekerjaan

Catatan

Kate jori

Suami

Nama

Alamat

Telepon

Kontak

Nama

Alamat

Telepon

Batal Simpan

Baru Ubah Hapus Aktif Cari Laporan Desain Keluar

Contoh-contoh laporan yang dihasilkan :

## RSI Darus Syifa Benowo

TANGGAL : 05 Januari 2005 Jam 11:00:57 S/D TANGGAL : 07 Januari 2005 Jam 11:00:57

| No.   | Tanggal    | Jam      | No Resep     | No. RM | Nama Pasien     | Jenis Pembelian | Nilai Barang | Jasa | Diskon    | Total        |
|-------|------------|----------|--------------|--------|-----------------|-----------------|--------------|------|-----------|--------------|
| 1     | 06/01/2005 | 01:17:28 | TO_ID/2005/1 |        | A FAUZI         | BEBAS           | 816,750.00   | .00  | .00       | 816,750.00   |
| 2     | 06/01/2005 | 20:32:05 | TO_ID/2005/2 | 069636 | HENDRO GOENAWAN | DISCOUNT15      | 75,920.00    | .00  | 11,388.00 | 75,315.50    |
| 3     | 06/01/2005 | 20:32:05 | TO_ID/2005/3 |        | SHANTI          | BEBAS           | 163,553.00   | .00  | .00       | 163,553.00   |
| 4     | 06/01/2005 | 21:12:40 | TO_ID/2005/4 |        | ABD.RASYID      | DISCOUNT15      | 48,125.00    | .00  | 7,219.80  | 40,906.30    |
| Total |            |          |              |        |                 |                 | 1,104,348.00 | .00  | 18,607.80 | 1,096,524.80 |



**KARTU INDUK UTAMA PASIEN**  
**RSI Darus Syifa Benowo**  
**PERIODE TANGGAL : 18-01-2005 S/D 28-01-2005**

Tanggal Kunjungan : 18 Januari 2005

| NO<br>URUT | Nama Poliklinik | Kasus Baru | Kasus Lama | Jumlah<br>Kunjungan |
|------------|-----------------|------------|------------|---------------------|
| 1          | GIGI            | 0          | 0          | 1                   |
| Total      |                 | 0          | 0          | 1                   |

Tanggal Kunjungan : 19 Januari 2005

| NO<br>URUT | Nama Poliklinik | Kasus Baru | Kasus Lama | Jumlah<br>Kunjungan |
|------------|-----------------|------------|------------|---------------------|
| 1          | BEDAH           | 0          | 0          | 1                   |
| 2          | GIGI            | 0          | 0          | 1                   |
| 3          | JANTUNG         | 0          | 0          | 1                   |
| 4          | KANDUNGAN       | 1          | 1          | 1                   |
| 5          | MATA            | 0          | 0          | 1                   |
| Total      |                 | 1          | 1          | 5                   |

Tanggal Kunjungan : 20 Januari 2005

| NO<br>URUT | Nama Poliklinik | Kasus Baru | Kasus Lama | Jumlah<br>Kunjungan |
|------------|-----------------|------------|------------|---------------------|
| 1          | GIGI            | 0          | 0          | 1                   |
| Total      |                 | 0          | 0          | 1                   |

Total Keseluruhan Kunjungan : 7

Jl. Raya Benowo  
Telp. (031) 1234435  
Surabaya

## BUKU REGISTER PENDAFTARAN PASIEN RAWAT JALAN

Periode : 18 Januari 2005 s.d. 20 Januari 2005

| Tanggal    | Nomor |                | N a m a              | A l a m a t      | Umur/<br>JKL |    | Kunjungan   | Cara Bayar | P o l i               |
|------------|-------|----------------|----------------------|------------------|--------------|----|-------------|------------|-----------------------|
|            | No    | Rekam<br>Medik |                      |                  | L            | P  |             |            |                       |
| 1.         | 2.    | 3.             | 4.                   | 5.               | 6.           | 7. | 8           | 9.         | 10                    |
| 18/01/2005 | 1     | 068328         | ABDUL HASAN          | RANDU PADANGAN   | 51           |    | PEMERIKSAAN | UMUM       | Poliklinik Gigi       |
| 19/01/2005 | 2     | 069638         | FIRMAN RAJASA        | JL. BOJONEGORO   | 25           |    | PEMERIKSAAN | UMUM       | Unit Gawat Darurat    |
| 19/01/2005 | 3     | 067227         | ABDUL ROSID          | NGASINAN         | 42           |    | PEMERIKSAAN | UMUM       | Poliklinik Mata       |
| 19/01/2005 | 4     | 068734         | AGUNG                | TSI CC/02        | 21           |    | PEMERIKSAAN | UMUM       | Poliklinik Gigi       |
| 19/01/2005 | 5     | 068585         | AKHMAD EFENDI        | NGASINAN NO.8    | 26           |    | PEMERIKSAAN | UMUM       | Poliklinik Bedah Umum |
| 19/01/2005 | 6     | 067376         | ADITYA EKA YULIANSAI | KANDANGAN        |              | 30 | PEMERIKSAAN | UMUM       | Poliklinik Kandungan  |
| 19/01/2005 | 7     | 068468         | ABDUL RAHMAN         | SEMEMI JAYA 7/25 | 41           |    | PEMERIKSAAN | UMUM       | Poliklinik Jantung    |
| 20/01/2005 | 8     | 069639         | HADI                 | JL. MALANG       |              | 25 | PEMERIKSAAN | UMUM       | Poliklinik Gigi       |



